Best Available Copy

(12) UK Patent Application (19) GB (11) 2 347 329 (13) A

(43) Date of A Publication 30.08.2000

(21) Application No 9930699.5

(22) Date of Filing 24.12.1999

(30) Priority Data
(31) 09221750 (32) 28.12.1998 (33) US

(51) INT CL[7]
G06F 17/30

(52) UK CL (Edition R)
H4P PPEC

(56) Documents Cited
EP 0969389 A2 EP 0718783 A1 WO 99/61984 A1
WO 99/35595 A2 WO 98/36344 A2

(58) Field of Search
UK CL (Edition) H4P PPA PPEC
INT CL[7] G06F 13/14 13/38 17/30
Online Databases: WPI, EPODOC, JAPIO

(71) Applicant(s)
Spyglass Inc
(Incorporated in USA - Delaware)
1240 E Diehl, Naperville, Illinois 60563,
United States of America

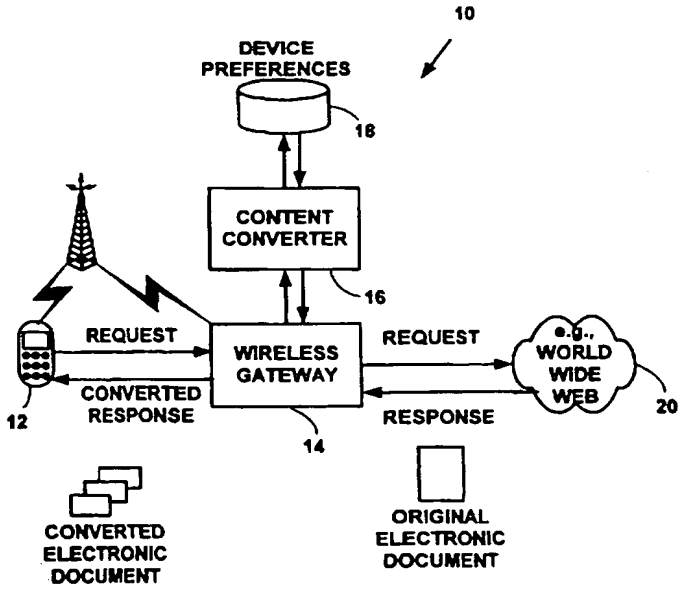
(72) Inventor(s)
Kenneth F Carlino
Kenneth B Hohl

(74) Agent and/or Address for Service
Frank B Dehn & Co
179 Queen Victoria Street, LONDON, EC4V 4EL,
United Kingdom

(54) Abstract Title
Converting electronic documents into a format suitable for a wireless device

(57) Electronic documents such as web pages requested from the Internet by a wireless device 12 are converted at a gateway 14 from a first markup language such as hypertext markup language (HTML) into a second markup language such as wireless markup language (WML) from a wireless application protocol (WAP). The electronic document is converted into a format suitable for display on the wireless device based on device type and preference information supplied by the wireless device. The electronic document is divided into textual elements and non-textual elements (e.g. images) before conversion. A converted document is created from the converted elements. Three separate methods of dividing and converting the original document are described. The invention is also applicable to wireline devices.

FIG. 1



GB 2 347 329 A

At least one drawing originally filed was informal and the print reproduced here is taken from a later filed formal copy.

FIG. 1

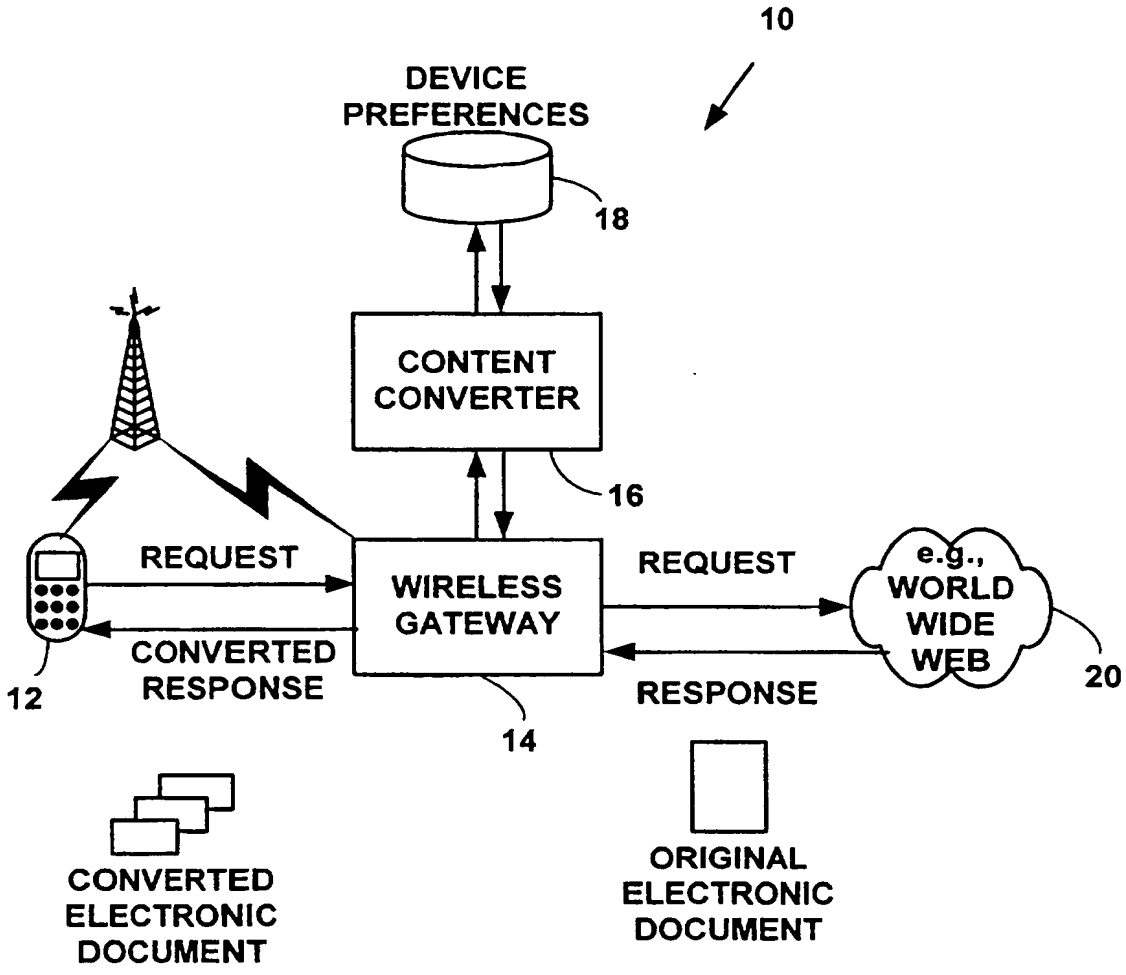

10
DEVICE PREFERENCES
18
CONTENT CONVERTER
16
REQUEST
WIRELESS GATEWAY
REQUEST
e.g., WORLD WIDE WEB
20
CONVERTED RESPONSE
RESPONSE
12
14
CONVERTED ELECTRONIC DOCUMENT
ORIGINAL ELECTRONIC DOCUMENT

FIG. 2

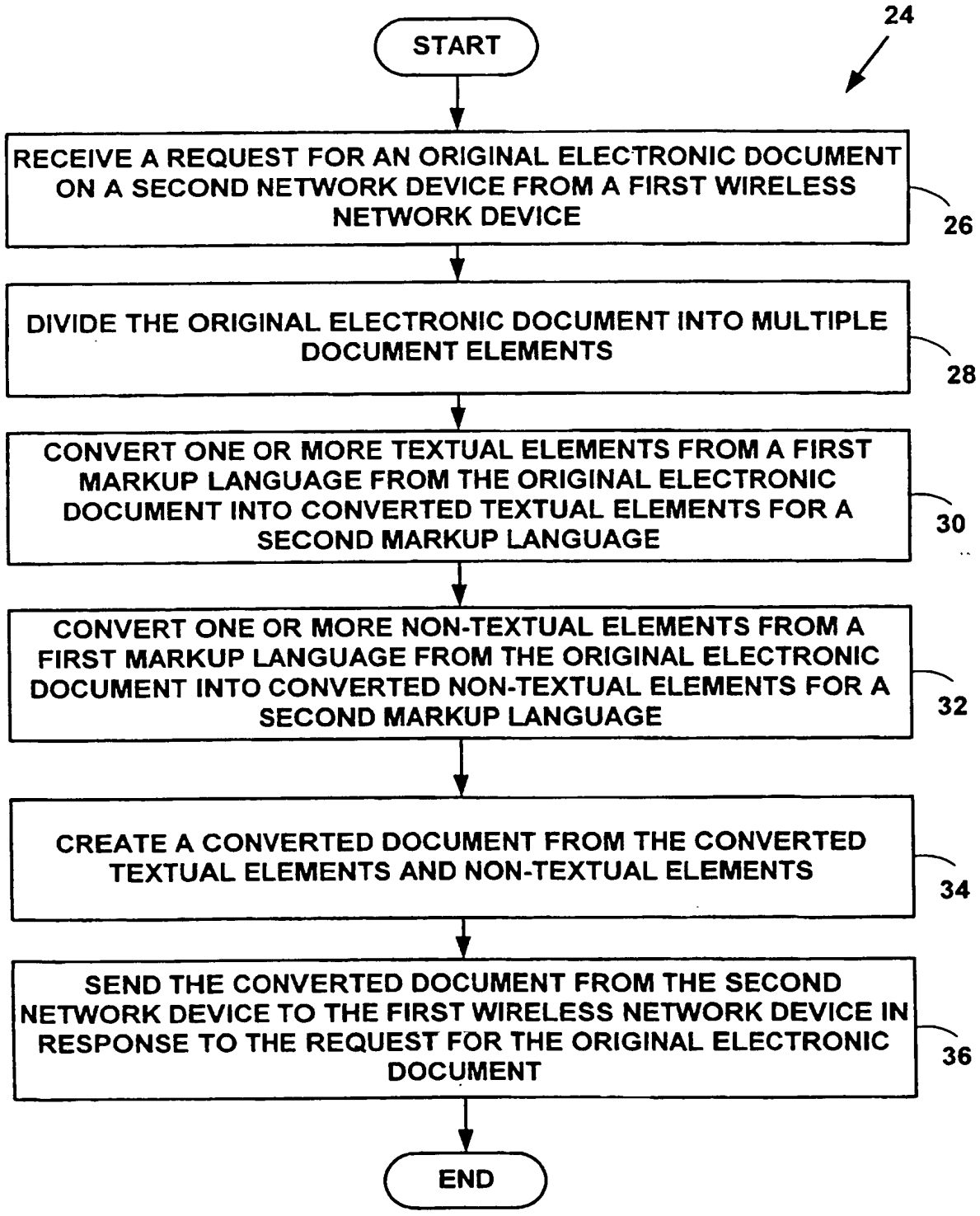

24
START
RECEIVE A REQUEST FOR AN ORIGINAL ELECTRONIC DOCUMENT ON A SECOND NETWORK DEVICE FROM A FIRST WIRELESS NETWORK DEVICE
26
DIVIDE THE ORIGINAL ELECTRONIC DOCUMENT INTO MULTIPLE DOCUMENT ELEMENTS
28
CONVERT ONE OR MORE TEXTUAL ELEMENTS FROM A FIRST MARKUP LANGUAGE FROM THE ORIGINAL ELECTRONIC DOCUMENT INTO CONVERTED TEXTUAL ELEMENTS FOR A SECOND MARKUP LANGUAGE
30
CONVERT ONE OR MORE NON-TEXTUAL ELEMENTS FROM A FIRST MARKUP LANGUAGE FROM THE ORIGINAL ELECTRONIC DOCUMENT INTO CONVERTED NON-TEXTUAL ELEMENTS FOR A SECOND MARKUP LANGUAGE
32
CREATE A CONVERTED DOCUMENT FROM THE CONVERTED TEXTUAL ELEMENTS AND NON-TEXTUAL ELEMENTS
34
SEND THE CONVERTED DOCUMENT FROM THE SECOND NETWORK DEVICE TO THE FIRST WIRELESS NETWORK DEVICE IN RESPONSE TO THE REQUEST FOR THE ORIGINAL ELECTRONIC DOCUMENT
36
END

FIG. 3

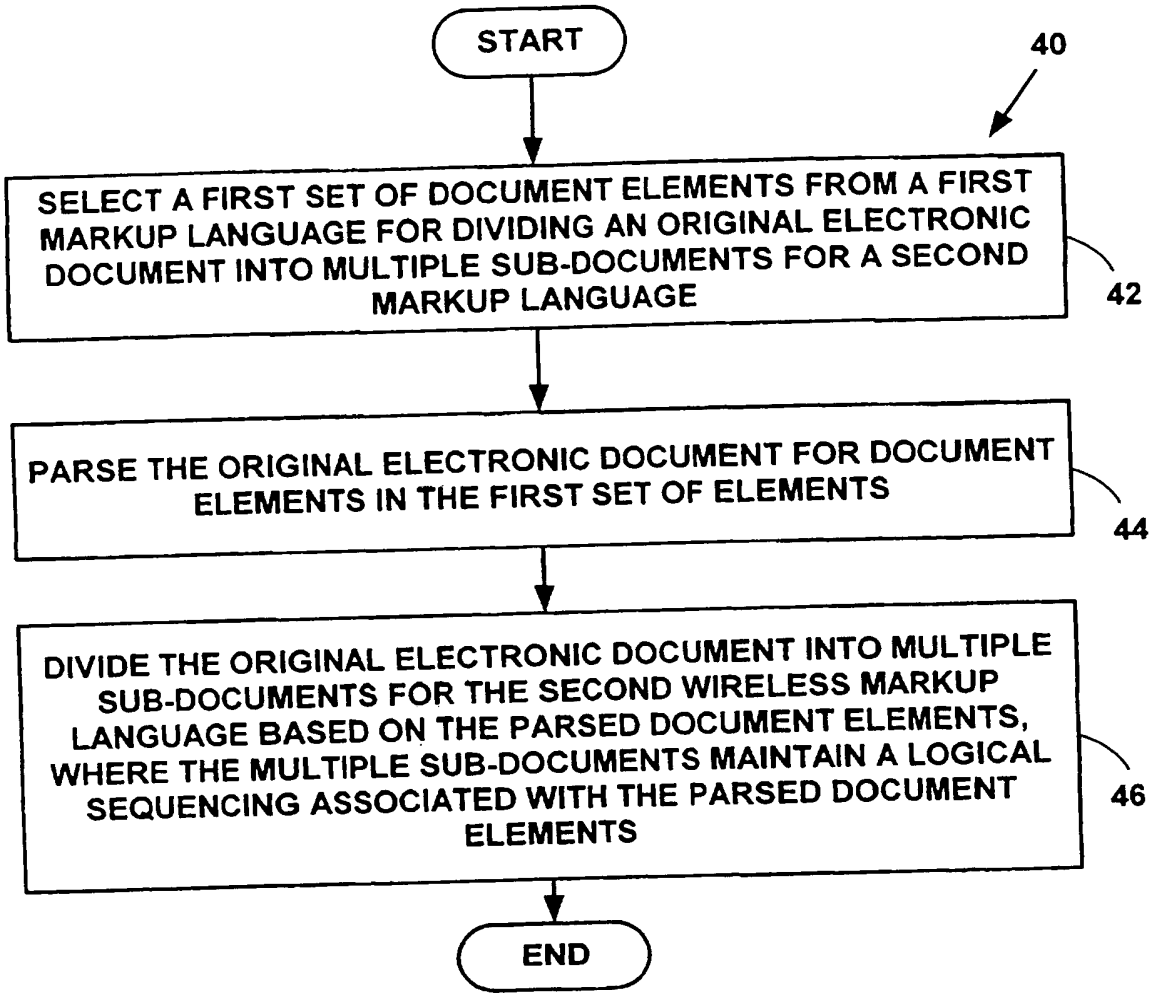

START
40
SELECT A FIRST SET OF DOCUMENT ELEMENTS FROM A FIRST MARKUP LANGUAGE FOR DIVIDING AN ORIGINAL ELECTRONIC DOCUMENT INTO MULTIPLE SUB-DOCUMENTS FOR A SECOND MARKUP LANGUAGE
42
PARSE THE ORIGINAL ELECTRONIC DOCUMENT FOR DOCUMENT ELEMENTS IN THE FIRST SET OF ELEMENTS
44
DIVIDE THE ORIGINAL ELECTRONIC DOCUMENT INTO MULTIPLE SUB-DOCUMENTS FOR THE SECOND WIRELESS MARKUP LANGUAGE BASED ON THE PARSED DOCUMENT ELEMENTS, WHERE THE MULTIPLE SUB-DOCUMENTS MAINTAIN A LOGICAL SEQUENCING ASSOCIATED WITH THE PARSED DOCUMENT ELEMENTS
46
END

FIG. 4

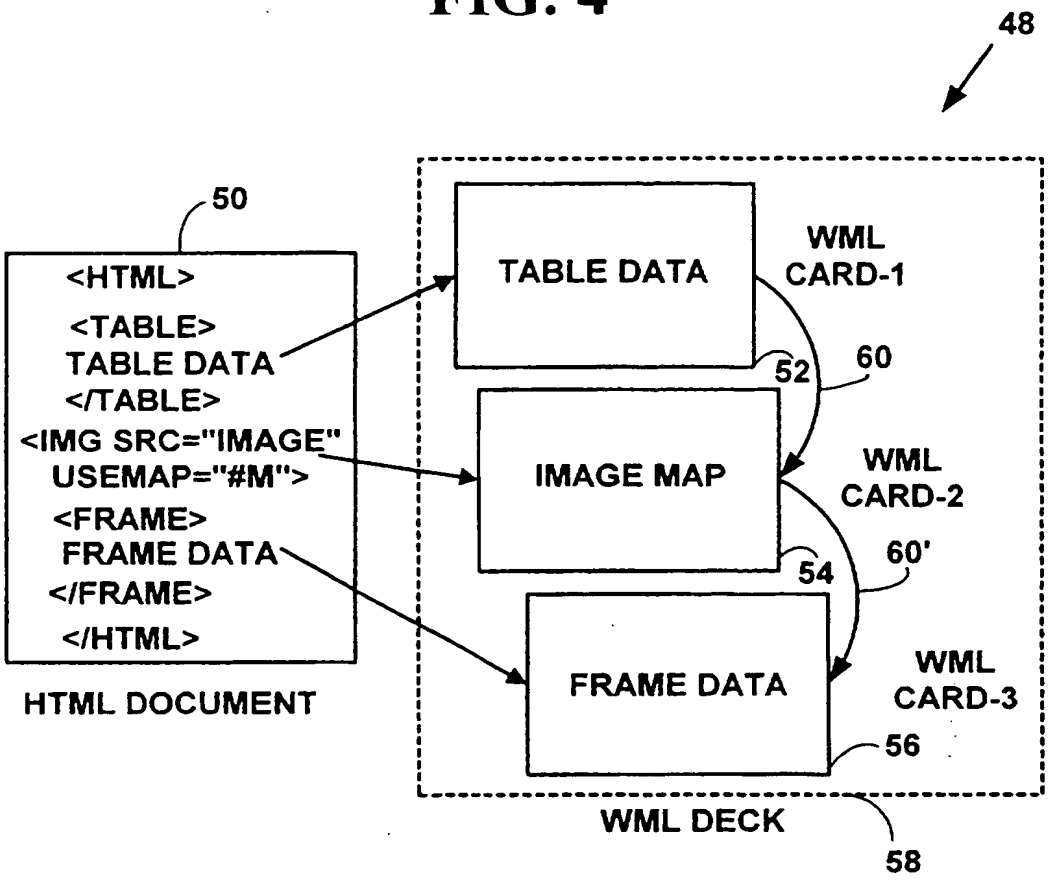

48
50
<HTML>
<TABLE>
TABLE DATA
</TABLE>
<IMG SRC="IMAGE"
USEMAP="#M">
<FRAME>
FRAME DATA
</FRAME>
</HTML>
HTML DOCUMENT
TABLE DATA
WML
CARD-1
52
60
IMAGE MAP
WML
CARD-2
54
60'
FRAME DATA
WML
CARD-3
56
WML DECK
58

FIG. 5

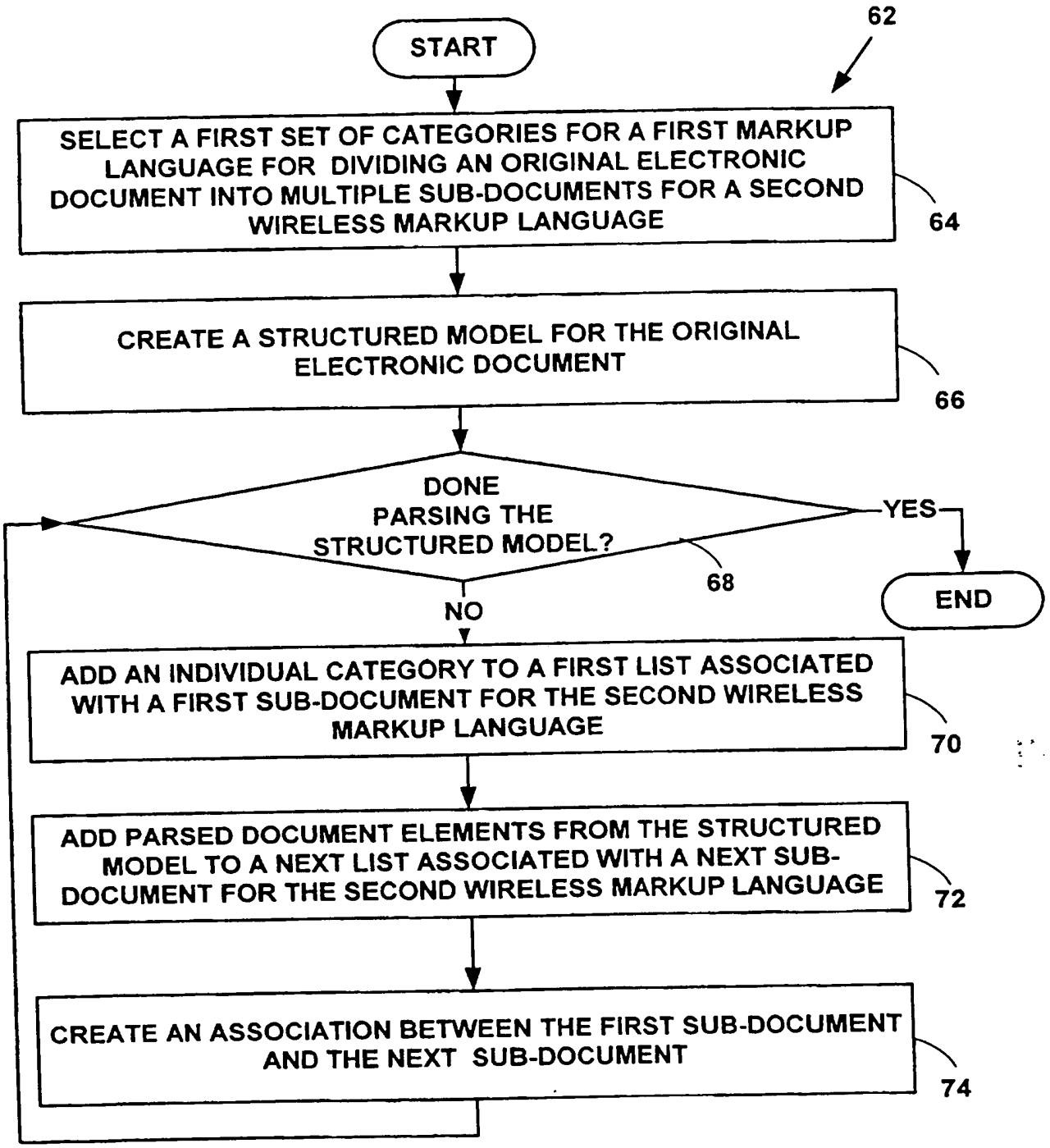

START
62
SELECT A FIRST SET OF CATEGORIES FOR A FIRST MARKUP LANGUAGE FOR DIVIDING AN ORIGINAL ELECTRONIC DOCUMENT INTO MULTIPLE SUB-DOCUMENTS FOR A SECOND WIRELESS MARKUP LANGUAGE
64
CREATE A STRUCTURED MODEL FOR THE ORIGINAL ELECTRONIC DOCUMENT
66
DONE PARSING THE STRUCTURED MODEL?
68
YES
END
NO
ADD AN INDIVIDUAL CATEGORY TO A FIRST LIST ASSOCIATED WITH A FIRST SUB-DOCUMENT FOR THE SECOND WIRELESS MARKUP LANGUAGE
70
ADD PARSED DOCUMENT ELEMENTS FROM THE STRUCTURED MODEL TO A NEXT LIST ASSOCIATED WITH A NEXT SUB-DOCUMENT FOR THE SECOND WIRELESS MARKUP LANGUAGE
72
CREATE AN ASSOCIATION BETWEEN THE FIRST SUB-DOCUMENT AND THE NEXT SUB-DOCUMENT
74

FIG. 6

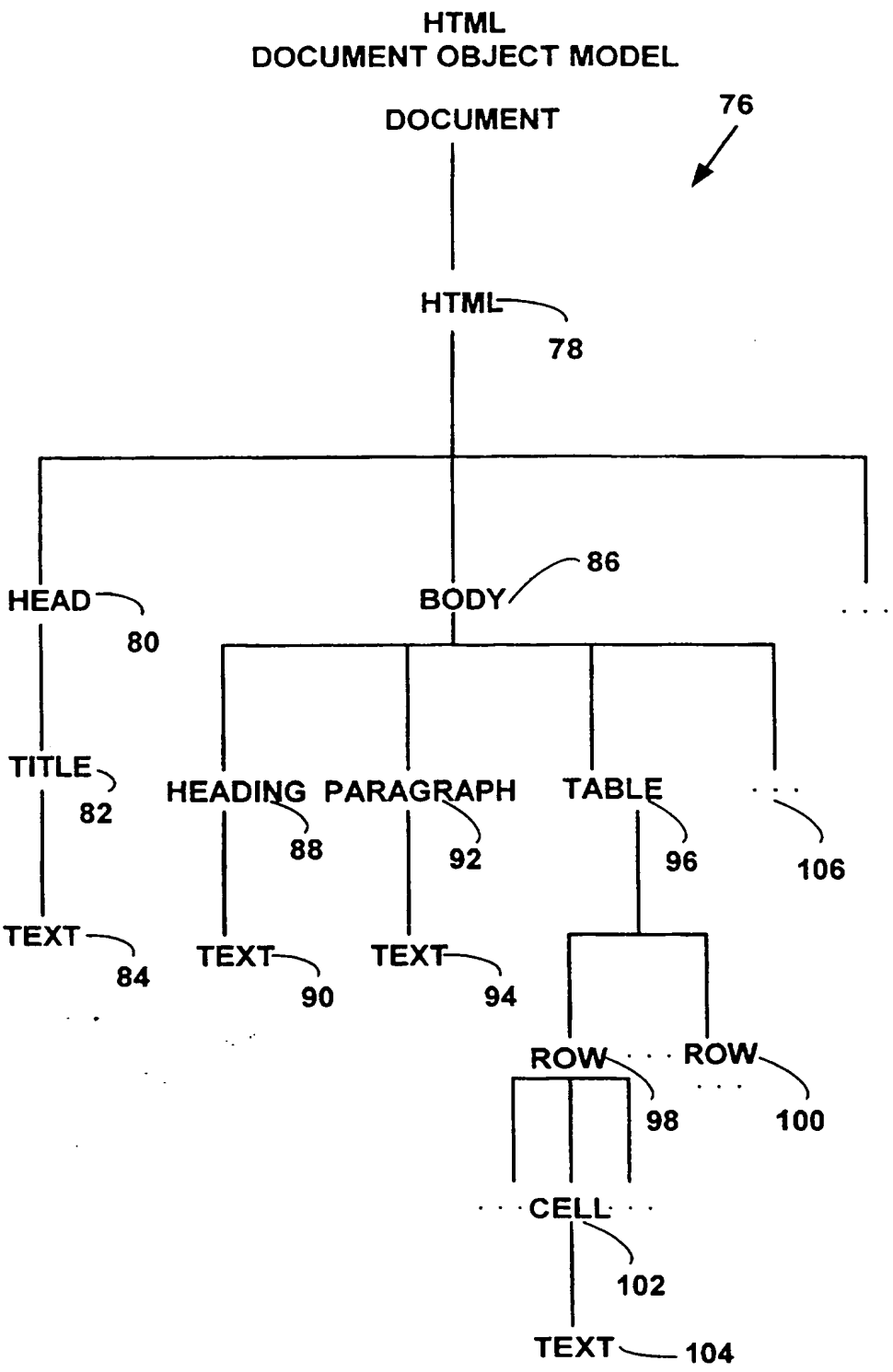

HTML
DOCUMENT OBJECT MODEL
DOCUMENT
76
HTML
78
HEAD
80
BODY
86
TITLE
82
HEADING
88
PARAGRAPH
92
TABLE
96
106
TEXT
84
TEXT
90
TEXT
94
ROW
98
ROW
100
CELL
102
TEXT
104

FIG. 7

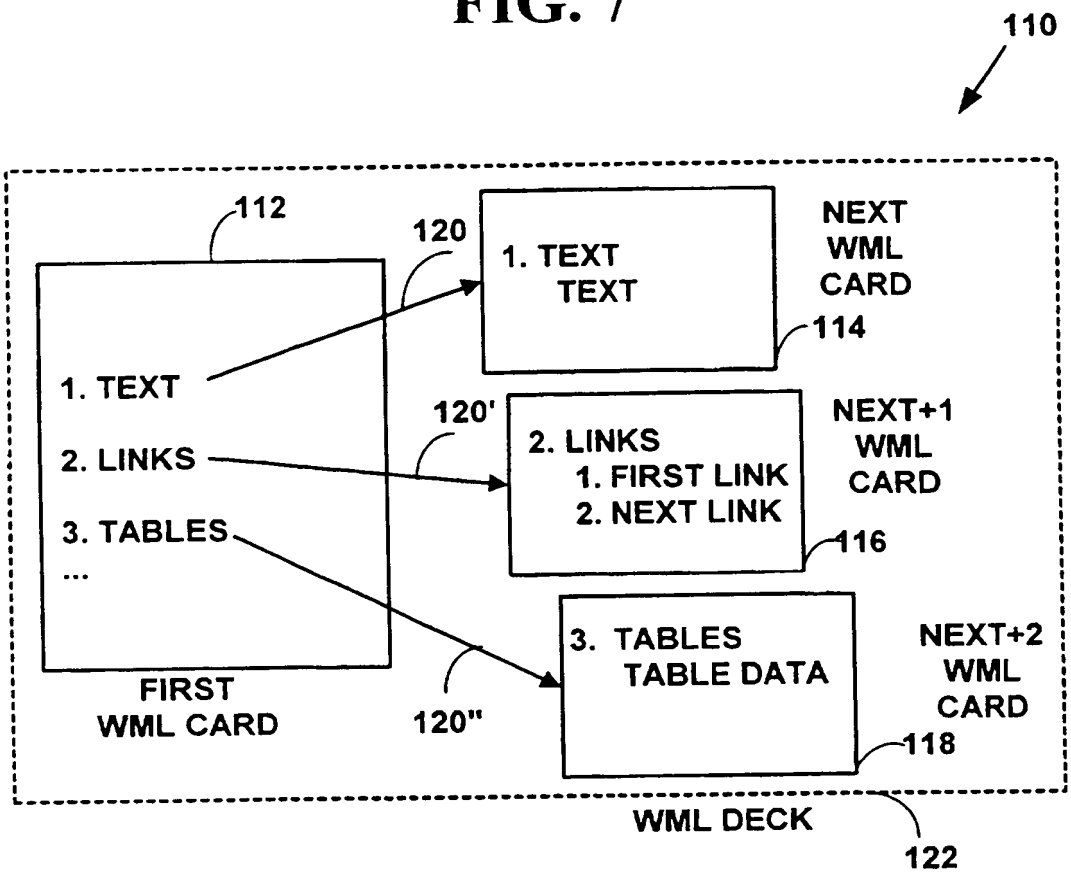

110
112
120
1. TEXT
TEXT
NEXT
WML
CARD
114
1. TEXT
120'
2. LINKS
1. FIRST LINK
2. NEXT LINK
NEXT+1
WML
CARD
116
2. LINKS
3. TABLES
...
3. TABLES
TABLE DATA
NEXT+2
WML
CARD
118
FIRST
WML CARD
120"
WML DECK
122

FIG. 8

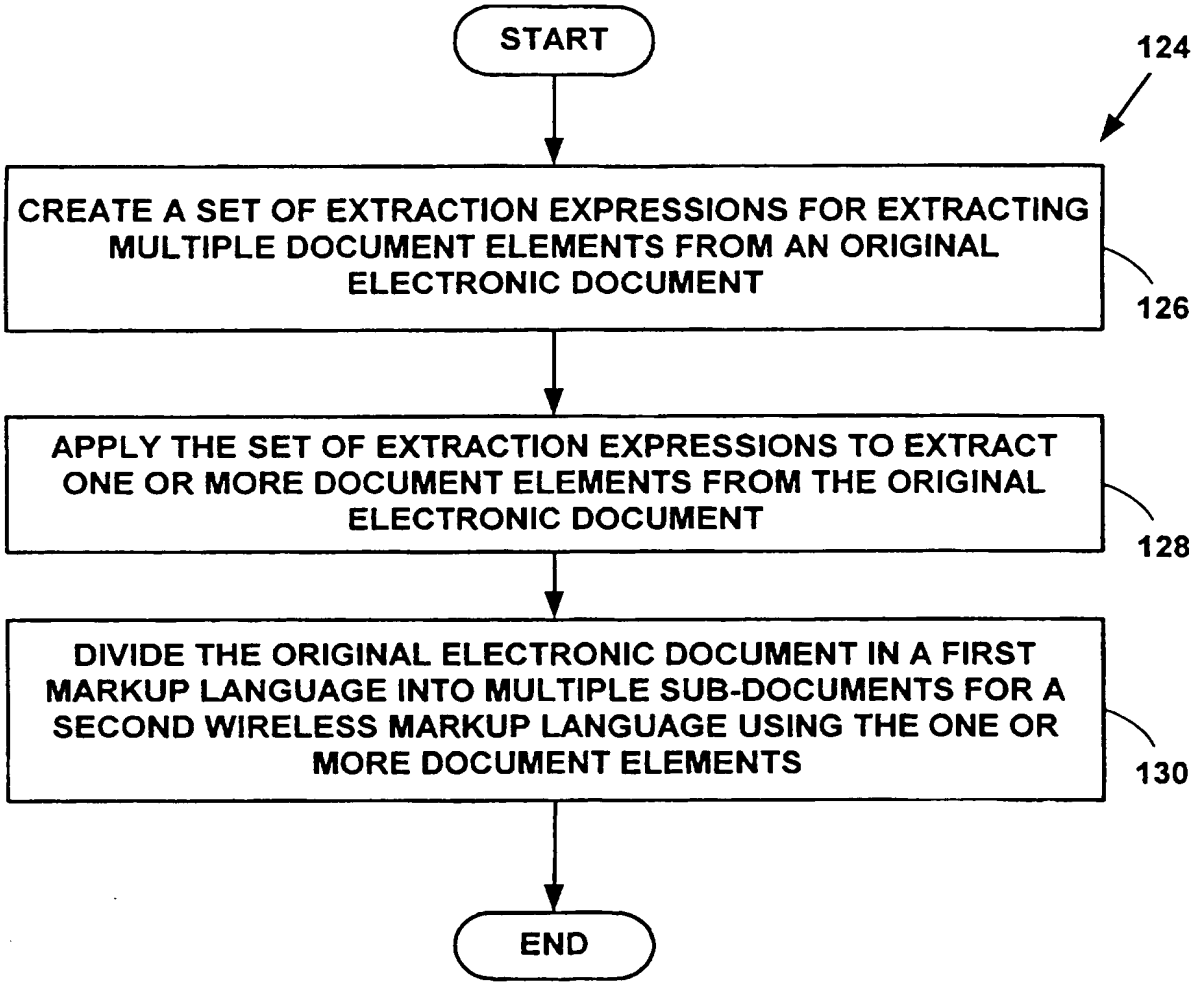

START
124
CREATE A SET OF EXTRACTION EXPRESSIONS FOR EXTRACTING MULTIPLE DOCUMENT ELEMENTS FROM AN ORIGINAL ELECTRONIC DOCUMENT
126
APPLY THE SET OF EXTRACTION EXPRESSIONS TO EXTRACT ONE OR MORE DOCUMENT ELEMENTS FROM THE ORIGINAL ELECTRONIC DOCUMENT
128
DIVIDE THE ORIGINAL ELECTRONIC DOCUMENT IN A FIRST MARKUP LANGUAGE INTO MULTIPLE SUB-DOCUMENTS FOR A SECOND WIRELESS MARKUP LANGUAGE USING THE ONE OR MORE DOCUMENT ELEMENTS
130
END

FIG. 9

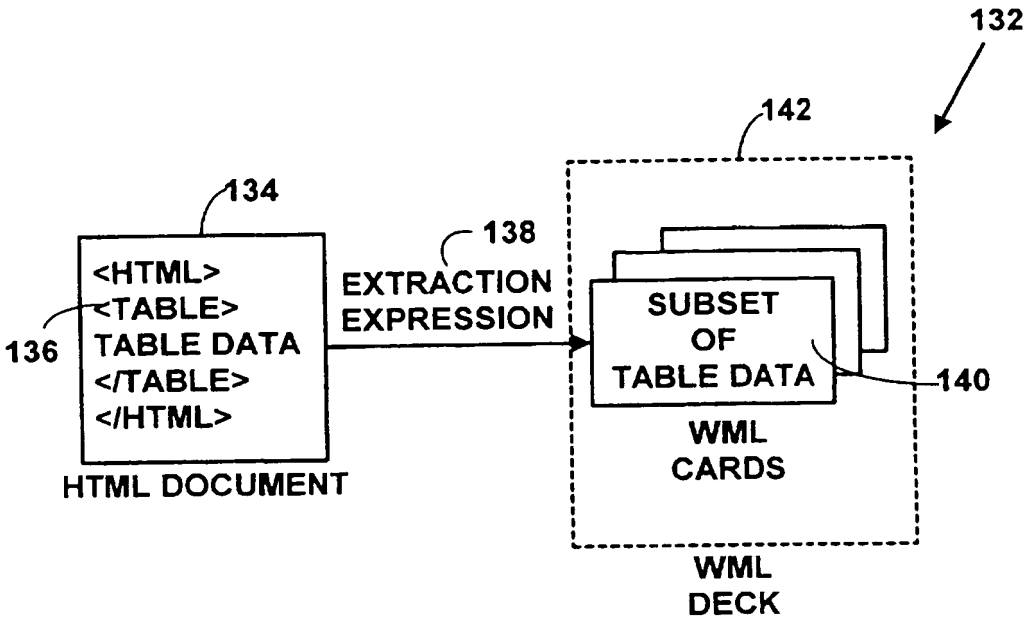

132
142
134
138
<HTML>
<TABLE>
TABLE DATA
</TABLE>
</HTML>
136
EXTRACTION EXPRESSION
SUBSET OF TABLE DATA
140
WML CARDS
HTML DOCUMENT
WML DECK

FIG. 10

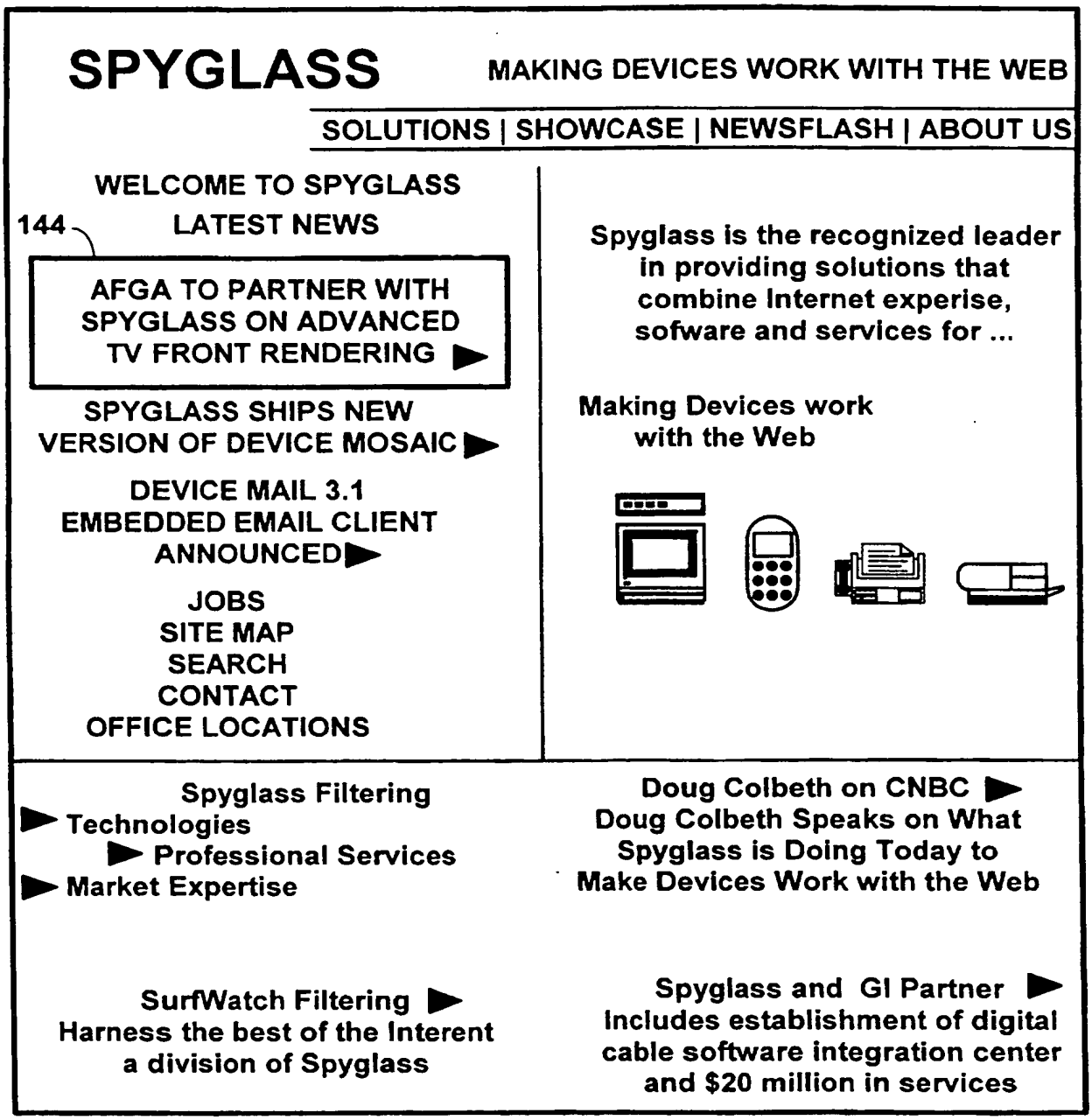
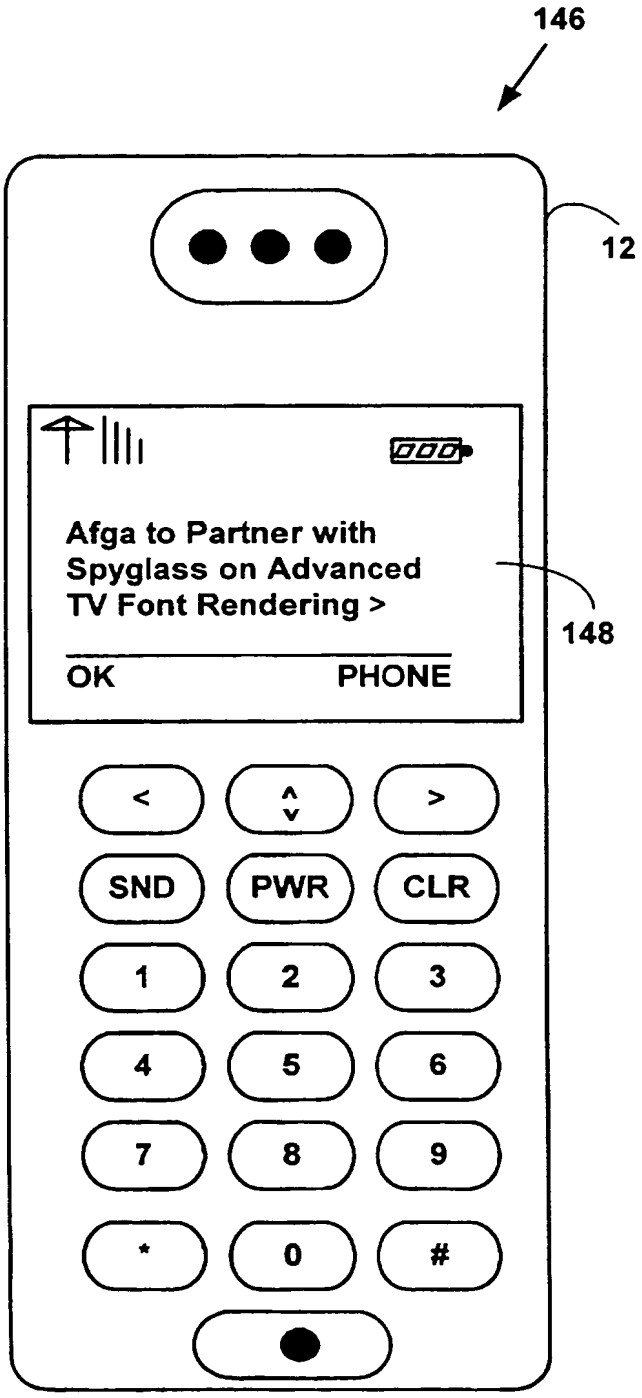

142
SPYGLASS
MAKING DEVICES WORK WITH THE WEB
SOLUTIONS | SHOWCASE | NEWSFLASH | ABOUT US
WELCOME TO SPYGLASS
144
LATEST NEWS
AFGA TO PARTNER WITH SPYGLASS ON ADVANCED TV FRONT RENDERING
SPYGLASS SHIPS NEW VERSION OF DEVICE MOSAIC
DEVICE MAIL 3.1 EMBEDDED EMAIL CLIENT ANNOUNCED
JOBS
SITE MAP
SEARCH
CONTACT
OFFICE LOCATIONS
Spyglass is the recognized leader in providing solutions that combine Internet experise, sofware and services for ...
Making Devices work with the Web
Spyglass Filtering Technologies
Professional Services
Market Expertise
Doug Colbeth on CNBC
Doug Colbeth Speaks on What Spyglass is Doing Today to Make Devices Work with the Web
SurfWatch Filtering
Harness the best of the Interent
a division of Spyglass
Spyglass and GI Partner
Includes establishment of digital cable software integration center and $20 million in services

FIG. 11


146
12
Afga to Partner with
Spyglass on Advanced
TV Font Rendering >
148
OK
PHONE
<
>
SND
PWR
CLR
1
2
3
4
5
6
7
8
9
*
0
#

Method and System for Converting Content of Electronic Data for Wireless Devices

A portion of the disclosure of this patent document contains material, which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent disclosure, as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all copyright rights whatsoever.

This invention relates to computer networks. More specifically, it relates to a method and system for converting the content of electronic data for wireless devices.

The Internet is a world-wide network of interconnected computers. The World-Wide-Web is an information system on the Internet designed for electronic document interchange. Electronic documents on the World-Wide-Web are typically stored in files that include text, hypertext, references to graphics, animation, audio, video and other electronic data. The structure of hypertext documents is defined by document markup languages such as Standard Generalized Markup Language ("SGML"), Hyper Text Markup Language ("HTML"), Compact Hyper Text Markup Language, eXtensible Markup Language ("XML"), Virtual Reality Markup Language ("VRML"), Voice extensible Markup Language, ("VoxML") and others.

As is known in the art, a hypertext document includes markup codes called "tags." Tags define the structure of a hypertext document and typically includes at least a "begin" tag name enclosed by a delimiter and, in many instances, an "end" tag name

enclosed by a delimiter. For example, the markup tag "<H1>" signifies the beginning of a Hyper Text Markup Language first level header, and the markup tag "</H1>" signifies the end of a Hyper Text Markup Language first level header. However, the Hyper Text Markup Language image tag "<IMG ...>" ends with the closing tag delimiter ">" and does not use an end tag in the format "</IMG>". Other markup languages have similar tags used to create hypertext documents.

Markup languages allow references to additional content besides text including graphics, animation, audio, video and other electronic data. For example, the Hyper Text Markup Language allows use of graphical images in a hypertext document with an image "<IMG>" tag. An exemplary Hyper Text Markup Language image tag <IMG SRC="logo.jpg"> allows a graphical logo image stored in a Joint Pictures Expert Group file "logo.jpg," to be displayed.

Hypertext documents from the World-Wide-Web are typically displayed for a user with a software application called a "browser" such as Internet Explorer, by Microsoft Corporation of Redmond Washington, Netscape Navigator, by Netscape Communications of Mountain View, California, and others. A browser typically parses a hypertext document and converts hypertext, including markup tags, into a visual display of text, graphics, animation, audio, video, etc., for display on a device such as a personal computer display.

Additional content is retrieved in a hypertext document from other sources using "hyperlink" references within hypertext documents. For example, an exemplary Hyper Text Markup Language hyperlink tag

"<A HREF="http://www.spyglass.com/logo.mov">" provides a hyperlink to a movie file "logo.mov." When a user selects the link (e.g., with a mouse click) in a hypertext document, the movie file "logo.mov" is located using a Uniform Resource Locator ("URL") from the location "www.spyglass.com." Hyper Text Transfer Protocol (e.g., "HTTP") is used as the transfer protocol.

Transfer protocols such as Hyper Text Transfer Protocol ("HTTP"), File Transfer Protocol ("FTP"), Gopher, and others provide a means for transferring hypertext documents or additional content from other locations on the World-Wide-Web. Hyper Text Transfer Protocol is one primary protocol used to transfer information on the World-Wide-Web. Hyper Text Transfer Protocol is a protocol that allows users to connect to a server, make a hypertext request, get a response, and then disconnect from the server.

File Transfer Protocol is a protocol that provides access to files on remote systems. Using File Transfer Protocol, a user logs onto a system, searches a directory structure and downloads or uploads a file. Gopher is a protocol similar to File Transfer Protocol. Gopher provides a series of menus linked to files containing actual hypertext.

Wireless devices, such as wireless phones, now have data capabilities in addition to voice capabilities. The data capabilities allow a wireless device to receive an electronic document from the World-Wide-Web. To optimize performance, and to provide an electronic document in a format useable on wireless devices, a Wireless Application Protocol ("WAP") is typically used. The Wireless Application Protocol includes several protocols and standards designed to provide wireless devices with access to an electronic document and was developed as an alternative to other markup languages and protocols developed for the World-Wide-Web. More information on the Wireless

Application Protocol can be found on the World-Wide-Web at the URL "http://www.wapforum.org."

One component of the Wireless Application Protocol is a Wireless Markup Language ("WML"), which includes markup tags, and provides control over formatting and layout of an electronic document. The Wireless Markup Language is often more appropriate to use for wireless devices such as wireless phones than other markup languages such as Hyper Text Markup Language.

Wireless Markup Language data is structured as a collection of "cards." A single collection of cards is referred to as a "deck." Each card includes structured content and navigation specifications. Logically, a user of a wireless device navigates through a series of cards, reviews the content of each card, enters requested information, selects options, and navigates to and from other cards in a deck.

A user with a wireless device may request an electronic document on the World-Wide-Web. However, there are several problems associated with displaying the content of an electronic document from the World-Wide-Web on a smaller display on a wireless device.

First, most electronic documents developed for the World-Wide-Web are based on the assumption of viewing with standard "SuperVGA" resolution (e.g., 800x600 pixel resolution with 256 or more colors). A user with a wireless device, such as wireless phone, may desire to view electronic documents from the World-Wide-Web. However, most wireless devices have a display with a resolution that is less than SuperVGA (e.g., 100x200 monochrome pixel resolution) typically does not support color. In addition, a SuperVGA device is typically capable of displaying 24 lines or more of 80 character or

more of text. A wireless device display typically can display about two to five lines of 20 characters of text. Thus, the content of electronic documents will not be properly displayed and may be difficult to view based on the original hypertext content. However, an electronic document should communicate the same content to devices with a SuperVGA display as well as to the wireless devices with a smaller display that is less than SuperVGA in resolution.

One solution is to store several versions of the electronic document at a content provider site. For example, one version of the electronic document is stored in the Hyper Text Markup Language and another in the Wireless Markup Language. However, this solution requires a large amount of storage space and creates a very difficult maintenance problem to keep the various versions of the electronic document synchronized when changes are made.

Another problem is that displays on wireless devices may not be capable of displaying, or have enough memory to download and display a full page of text, graphical images, animation, video or other content included in an electronic document. Trying to display an electronic document with too much text, graphical images, animation or video may overload or otherwise adversely affect the wireless device leading to user frustration.

Yet another problem is that many users of wireless devices will not be using their wireless device for "general" browsing of the World-Wide Web. Instead, a wireless device user will typically be looking for specific information such as phone numbers, addresses, stock quotes, sports scores, current news, etc. Thus, such users may desire to have some types of electronic context from a hyper text document not displayed at all (e.g., no graphical images).

Thus, it is desirable to provide a method and system to convert an original electronic document into a converted electronic document useable on a wireless device such as a wireless phone. The method and system should allow virtually any electronic document on the World-Wide-Web to be converted and displayed in a format appropriate for a wireless device.

In accordance with preferred embodiments of the present invention, some of the problems associated with displaying electronic documents on a wireless device are overcome. A method and system for converting content of electronic documents is provided. One aspect of the invention includes a method for converting content of electronic documents. The method includes receiving a request for an original electronic document on a second network device on a first network from a first wireless device on the first network. The request includes an indication of the type of wireless device. An original electronic document for a first markup language is divided into multiple document elements including textual elements and non-textual elements for conversion into a second wireless markup language. One or more original textual elements from the original electronic document in the first markup language are converted into converted textual elements for a second wireless markup language. One or more original non-textual elements are converted from an original non-textual format for the first markup language into a converted non-textual format for the second wireless markup language. A converted electronic document is created from the converted textual elements and the converted non-textual elements based on the device type for the first wireless device from the request. The converted electronic document is sent from the second network device to the first wireless device in response to the request for the original electronic document. The present invention is not limited to wireless devices, and other network devices could also be used in place of the first wireless device.

Another aspect of the invention includes a system for content conversion. The system includes a wireless gateway for accepting requests for original electronic

documents from network devices such as wireless devices. A content converter application converts an original electronic document in a first markup language into a second converted document in a second wireless markup language suitable for display on a wireless device. A database stores conversion preferences for multiple wireless devices. The conversion preferences are used to further convert an original electronic document in a first markup language into a second converted document in a second wireless markup language suitable for display on a specific type of wireless device. However, more or fewer system components can also be used, and the present invention is not limited to the system components described. In addition, The system is not limited to wireless devices, and other network devices could also be used in place of the first wireless device.

In one exemplary preferred embodiment of the present invention, the method and system are used to allow a wireless device to request an original electronic document (e.g., a Hyper Text Markup Language document) and receive a converted electronic document (e.g., a Wireless Markup Language document) suitable for display on the wireless device. The converted electronic document suitable for display on a wireless device may provide increased satisfaction for a user of a wireless device.

For example, an original electronic document with the current weather may be requested from a wireless device. The original electronic document includes several graphical images of current weather maps and text for the current weather. The method and system may provide a converted electronic document including only the text for the current weather in a format suitable for display on the wireless device. However, the

present invention is not limited to wireless devices and other network devices could also be used.

Preferred embodiments of the invention will now be described, by way of example only, and with reference to the accompanying drawings in which:

FIG. 1 is a block diagram illustrating an exemplary content conversion system;

FIG. 2 is a flow diagram illustrating a method for content conversion;

FIG. 3 is a flow diagram illustrating a method for logical division of electronic document elements;

FIG. 4 is a block diagram visually illustrating the method for logical division of electronic document elements of FIG. 3;

FIG. 5 is a flow diagram illustrating a method for categorized division of electronic document elements;

FIG. 6 is a block diagram illustrating a portion of an exemplary Document Object Model ("DOM") for Hyper Text Markup Language;

FIG. 7 is a block diagram visually illustrating the method for categorized division of electronic document elements of FIG. 5;

FIG. 8 is a flow diagram illustrating a method for targeted conversions of electronic document elements;

FIG. 9 is block diagram visually illustrating the method for targeted conversion of electronic document elements of FIG. 8;

FIG. 10 is screen display illustrating an exemplary electronic document from the World-Wide-Web; and

FIG. 11 is a block diagram illustrating exemplary converted output from the screen display of FIG. 10 on an exemplary wireless device.

Exemplary content conversion system

FIG. 1 is a block diagram illustrating an exemplary content conversion system 10 for one preferred embodiment of the present invention. Content conversion system 10 includes multiple components. A first wireless device 12 requests original electronic documents. The first wireless device 12 includes a display that is typically less that SuperVGA resolution (i.e., less than 800x600 pixel resolution or less than 256 colors).

As is known in the art and is described above, an electronic document includes text, hypertext, graphical data or references to graphical data images, audio, video and other content. A hypertext document includes markup codes called "tags." The structure of hypertext documents is defined by document markup languages such as Standard Generalized Markup Language ("SGML"), Hyper Text Markup Language ("HTML"), compact HTML ("cHTML"), eXtensible Markup Language ("XML"), Virtual Reality Markup Language ("VRML"), and voice extensible Markup Language ("VoxML"), others. Markup languages also allow references to additional content besides text including graphics, animation, audio, video and other electronic data.

Also described above, electronic documents are typically displayed for a user with a software application called a "browser." A browser on a hand-held device or wireless device may be a sub-set of a larger browser, and may not be capable of displaying complete content of a requested electronic document as stored on an electronic document server. A browser typically reads an electronic document and renders the electronic document content into a presentation of text, graphics, animation, audio, video, etc., for

display on a device such as a personal computer. Most electronic documents developed for the World Wide Web assume that users will view the content of the electronic document with a browser on a desktop computer screen with a standard "SuperVGA" resolution (e.g., 800x600 pixel resolution with 256 or more available colors).

Returning to FIG.1, a wireless gateway 14 services requests for electronic documents from the first wireless device 12 and sends converted electronic documents back to the first wireless device 12. A content converter application 16 converts content of an electronic document to a format usable on the first wireless device 12 (e.g., a device with a smaller display). A database 18 stores wireless device conversion preferences. FIG. 1 illustrates a single database 18. However, multiple database components can also be used. A computer network 20 provides one or more electronic document servers for supplying electronic documents.

In a preferred embodiment of the present invention, the computer network 20 is the World-Wide-Web on the Internet. As is known in the art, the Internet is a world-wide network of interconnected computers. The World-Wide-Web is an information system on the Internet designed for electronic document interchange. However, other computer networks with electronic document servers could also be used (e.g., an intranet or other Local Area Network "LAN").

FIG. 1 illustrates the content converter application 16 as an individual component. However, the content converter 16 can also be integral to the wireless gateway 14 (not illustrated in FIG. 1). A preferred embodiment of the present invention is not limited to the network components shown in the content conversion system 10 and more or fewer network components may also be used. In addition, the content conversion system 10

illustrates a first wireless device 12. In another embodiment of the present invention, the first wireless device 12 can be replaced with a wireline network device (i.e., a network device connected to a network with a temporary or permanent connection using a wire or cable). The wireline network devices would be capable of using one or more wireless markup languages (e.g., Wireless Markup Language).

In one preferred exemplary a preferred embodiment of the present invention, the wireless gateway 14 and the content converter application 16 are software components on a proxy server. However, the functionality of components from FIG. 1 can also be provided with a combination of hardware and software components, or as hardware components.

In one preferred embodiment of the present invention, the functionality of components from FIG. 1 is provided with software using object-oriented programming techniques and the C++ programming language. However, other object-oriented programming languages besides C++ could also be used. In addition, in a preferred embodiment of the present invention, the functionality of components of FIG. 1 can also be provided with non-object oriented programming languages (e.g., C programming language).

As is known in the art, object-oriented programming is used to design computer software including object-oriented objects that are easy to create, cost effective to modify, and reusable. Object-oriented objects include "object data" and "object services." Object services are provided through "object methods" (also called "object operations" or "object functions"). Object methods typically operate on private data such as "instance data" or "object state data" that an object owns. A collection of objects is

called an "object class" which is sometimes called an "object type." An object class acts as a template that describes the behavior of sets of objects. An object's implementation is typically encapsulated, and is hidden from public view. Object private instance data can only be accessed by object methods of an object class. Object public instance data is accessed through a public "object interface."

An operating environment for components of content conversion system 10 of a preferred embodiment the present invention includes a processing system with at least one high speed Central Processing Unit ("CPU") and a memory system. In accordance with the practices of persons skilled in the art of computer programming, the present invention is described below with reference to acts and symbolic representations of operations and instructions that are performed by the processing system, unless indicated otherwise. Such acts and operations are referred to as being "computer-executed" or "CPU executed." Although described with one CPU, alternatively multiple CPUs may be used for a preferred embodiment of the present invention.

The memory system may include main memory and secondary storage. The main memory is high-speed random access memory ("RAM"). Main memory can include any additional or alternative high-speed memory device or memory circuitry. Secondary storage takes the form of persistent long term storage, such as Read Only Memory ("ROM"), optical or magnetic disks, organic memory or any other volatile or non-volatile mass storage system. Those skilled in the art will recognize that the memory system can comprise a variety and/or combination of alternative components.

Acts and symbolically represented operations or instructions include the manipulation of electrical or biological signals by the CPU. The electrical or biological

signals cause transformation of data bits. The maintenance of data bits at memory locations in a memory system thereby reconfigures or otherwise alters the CPU's operation. The memory locations where data bits are maintained are physical locations that have particular electrical, magnetic, optical, or organic properties corresponding to the data bits.

The data bits may also be maintained on a computer readable medium including magnetic disks, optical disks, organic disks and any other volatile or non-volatile mass storage system readable by the CPU. The computer readable medium includes cooperating or interconnected computer readable medium, which exist exclusively on the processing system or may be distributed among multiple interconnected processing systems that may be local or remote to the processing system.

Wireless network devices and other network devices for preferred embodiments of the present invention include wireless network devices that can interact with content conversion system 10 with based on standards proposed by the Institute of Electrical and Electronic Engineers ("IEEE"), International Telecommunications Union-Telecommunication Standardization Sector ("ITU"), Internet Engineering Task Force ("IETF"), Wireless Application Protocol Forum ("WAP") Forum, or others. IEEE standards can be found on the World Wide Web at the Universal Resource Locator ("URL") "www.ieee.org." The ITU, (formerly known as the CCITT) standards can be found at the URL "www.itu.ch." IETF standards can be found at the URL "www.ietf.org." The WAP Forum standards can be found at the URL "www.wapforum.org."

Content conversion for wireless devices

FIG. 2 is flow diagram illustrating a Method 24 for content conversion. The Method 24 includes receiving a request for an original electronic document on a second network device 14 on a first network from a first wireless device 12 on the first network at Step 26. The request indicates a device type for the first wireless device 12. At Step 28, the original electronic document for a first markup language is divided into multiple document elements including textual elements and non-textual elements for conversion into a second wireless markup language. At Step 30, one or more of the original the textual elements from the original electronic document in the first markup language are converted into converted textual elements for the second wireless markup language. At Step 32, one or more of the original the non-textual elements are converted from an original non-textual format for the first markup language into a converted non-textual format for the second wireless markup language. At Step 34, a converted electronic document is created from the converted textual elements and the converted non-textual elements. At Step 36, the converted electronic document is sent from the second network device to the first wireless device in response to the request for the original electronic document. However, the present invention is not limited to wireless devices and other wireline network device that are capable of using the WAP could also be used in place of the first wireless device.

In one exemplary preferred embodiment of the present invention, the content converter application 16 on the wireless gateway 14 receives the request for the original electronic document. In such an embodiment of the present invention, the content converter application 16 requests the original electronic document from the computer network 20 (e.g., the World-Wide-Web). In another embodiment of the present

invention, the content converter application 16 retrieves a copy of the original electronic document from storage (e.g., volatile or non-volatile) such as a cache, associated with the wireless gateway 14. In such an embodiment, a copy of an original electronic document is stored in storage associated with the wireless gateway 14 after a first request by the first wireless network device 12.

In one exemplary preferred embodiment of the present invention, the request received at step 26 is a Hyper Text Transfer Protocol ("HTTP") request. The request includes a device type (e.g., in an HTTP request header) for the first wireless device 12. However, other requests can also be used (e.g., File Transfer Protocol ("FTP"), etc.).

As is known in the art, HTTP is a transfer protocol used to transfer data from an electronic document server on the World-Wide-Web. For more information on HTTP see Internet Engineering Task Force ("IETF") Request For Comments ("RFC") 2068, incorporated herein by reference.

At Step 28, the original electronic document is divided into multiple document elements including textual elements and non-textual elements for a first markup language for conversion into a second wireless markup language. In one exemplary preferred embodiment of the present invention, the first markup language is Hyper Text Markup Language ("HTML"). However, other markup languages could also be used, (e.g., XML, cHTML, SGML, VRML, VoxML, etc.). For more information on HTML see RFC-1866, incorporated herein by reference.

In one exemplary preferred embodiment of the present invention, the second wireless markup language is Wireless Markup Language ("WML") from Wireless Application Protocol ("WAP"). However, other wireless markup languages could also be

used. The WAP includes the WML as well as protocols for converting non-textual elements (e.g., images). For more information on WML see "Wireless Application Protocol Specification", by the Wireless Application Protocol Forum, April 1998, incorporated herein by reference. The Wireless Application Protocol Specification can be found at the URL "www.wapforum.org." However, other markup languages could also be used for the second wireless markup language.

In preferred embodiments of the present invention, three separate methods are used to divide or divide and convert the original electronic document into multiple textual elements and non-textual elements in the first markup language for conversion into textual elements and non-textual elements in a second wireless markup language at Step 28. However, more or fewer methods can also be used to divide the original electronic document into multiple textual and non-textual elements. The three division methods include logical division of electronic document elements, categorized division of electronic document elements, and targeted conversion of electronic document elements. The three separate division methods used at Step 28 will be explained below.

Table 1 illustrates exemplary pseudo-code for pre-processing an original electronic document received at Step 26, before it is divided at Step 28 for certain exemplary preferred embodiments of the present invention. However, the present invention is not limited to the pre-processing pseudo-code illustrated and Table 1, and other methods can be used to pre-process an original electronic document. The exemplary pseudo-code produces a token list that can be used to divide and/or divide and convert the original electronic document into a converted electronic document at Step 28.

Exemplary data structures used with the exemplary pseudo-code to manipulate the token list from Table 1 at Step 28 are illustrated in below.

```
© 1998 by Spyglass, Inc.

Initialize a conversion as follows:

create a new tag actions dictionary and markup tag hash table
for each item in tag translation data table
        insert the tag translation data table item into the tag actions dictionary

for each item in attr whitelist table
        if prior tag name is same as this item's tag name
                insert attr into new attr hash set
        else
                if prior tag name not empty
                        insert the new attr hash set into the tag actions dictionary
                else
                        create a new attr hash set
                        insert attr into new attr hash set
                        set prior tag name to this item's tag name
insert the last new attr hash set into the tag actions dictionary


Pre-process the original electronic document as follows:

parse the incoming document and create a linked list of tokens representing the document's data
and original markup language elements

modify the token list so that it represents syntactically correct and well-formed original markup
language elements


start at head of list
while there are still more document elements in list

        if token is openStartTag
                copy token name to currentTag
                if currentTag is one of special content tags
                        set corresponding special content flag to true

                lookup currentTag in tag actions dictionary
                        if not found
                                if currentTag not found in markup tag hash table
                                        set current token to prev token
                                        delete all tokens from next token through next
closeStartTag token
                                else
                                        delete all tokens (tag attributes) from next token through
next closeStartTag token

                else if action for tag is removeWithoutContent
```

```
                    set current token to prev token
                    delete all tokens from next token through next closeStartTag token

            else if action for tag is removeWithContent
                    set current token to prev token
                    delete all tokens from next token through matching endTag

            else if action for tag is replace
                    if replacing end tag only
                            set current token to prev token
                            delete all tokens from next token through next closeStartTag
token
                    if replacing start tag only
                            replace name of current token with replacement name
                            delete all tokens from next token up to next closeStartTag token

            else if action for tag is remove all attrs except for white-list
                    set current token to next token
                    while token is not a closeStartTag
                            copy token name to attr
                            lookup attr in hash set for currentTag
                            if not found
                                    set current token to prev token
                                    delete next token
                            set current token to next token

        else if token is endTag
                copy token name to currentTag
                lookup currentTag in tag actions dictionary
                if not found
                        set current token to prev token
                        delete next token
                if found
                        if action for tag is removeWithoutContent
                                set current token to prev token
                                delete next token

                        if action for tag is replace
                                if replacing start tag only
                                        set current token to prev token
                                        delete next token
                                if replacing end tag only
                                        replace type of current token with openStartTag
                                        create a new closeStartTag token and insert it in list after
current token
                                        replace name of current token with replacement name

        set current token to next token
```

Table 1.

At Step 30, one or more of the original textual elements from the original electronic document in the first markup language are converted into converted textual

elements for a second wireless markup language (e.g., using the token list created with the exemplary pseudo-code in Table 1). In one exemplary preferred embodiment of the present invention, the textual elements include text and hypertext, including hypertext markup codes. However, other textual elements could also be used and the present invention is not limited to text and hypertext elements.

Table 2 illustrates exemplary conversions used to convert textual elements for the first markup language into converted textual elements for the second wireless markup language at Step 30. However, more or fewer conversions could also be used, and the present invention is not limited to the conversions listed in Table 2. Not all of the conversions for Table 2 may be used for an original electronic document in a selected original markup language. Only a subset of the conversion operations may be supported for a selected original markup language.

Textual Conversion	Description

Textual clipping	The size of the original electronic document may not be suitable, even in a converted form, for delivery to a wireless device due to memory and bandwidth restrictions. To reduce the amount of text, the content converter application 16 will deliver a specified number of initial characters or paragraphs, with links to the clipped text. The links to the clipped text are accomplished with a virtual URL scheme. This conversion can be used conditionally for large blocks of text defined to be larger than a specified number of paragraph elements (e.g., HTML <P>).
Removing unsupported textual elements	Elements in the first markup language (e.g., HTML) that are not supported in the second wireless markup language (e.g., WML) are removed (e.g., HTML APPLET, BGSOUND, STRIKE, etc.). Non-essential elements in the first markup language (e.g., HTML COMMENTS and META elements, etc.) are removed to reduce the size of the converted document that will be created, since such elements are typically not essential to the structure or layout of the original content. This may be done during the preprocessing described in Table 1.
Removing unsupported textual element tags	Elements in the first markup languages that have no equivalent in the second wireless markup language are removed (e.g., HTML BODY, ACRONYM, BANNER, etc.). Any content between removed start and end element tags is not removed. This may be done during the preprocessing described in Table 1.

Replacing equivalent textual markup elements	Elements and attributes of the first markup language that have equivalent elements and attributes in the second wireless markup language are replaced with the appropriate element and/or attribute. (e.g., HR in HMTL is replaced with BR in WML) Some elements and attributes of the first markup language may map into multiple elements in the second wireless markup language, and visa versa. Some elements and attributes of the first and second wireless markup language may also be identical (e.g., B, I, U, for bold, italic or underline text respectively). This may be done during the preprocessing described in Table 1.
Adding textual markup elements	Textual markup elements that are required by the second wireless markup language, but do not exist in the first markup language are added. (e.g., for HTML to WML conversion: • If a DOCTYPE element does not exist, one appropriate for WML is added. If one does exist, it is translated. • An ACCESS PUBLIC="TRUE" element is added to the WML so that other WML documents can be linked to the converted document. • Markup elements that divide a document into WML cards is added.)
Removing unsupported attributes	Attributes in the first markup language that have no equivalents in the second wireless markup language, or that are associated with elements that are replaced with elements for the second wireless markup language that have no equivalent attributes are removed (e.g., HTML IMG, ALIGN, BORDER, attributes). This may be done during the preprocessing described in Table 1.
Converting form elements	Only select form elements from the first markup language have equivalent form elements in the second wireless markup language. The equivalent form elements are converted. The remaining form elements are removed. (e.g., the HTML SELECT element, and text and password INPUT elements are converted to WML, other HTML form elements are removed). If input fields in the form cannot be converted, then the entire form may be deleted since most server-based programs that process forms require input from all the original input fields for the form element to be filled correctly.

Converting table elements	Table related elements in the first markup language may not have equivalent table elements in the second wireless markup language. However, the relationship between rows and columns in the tables is preserved. Simple and complex tables are converted differently. Simple tables include only textual markup (e.g., HTML B, I, U, EM, STRONG, BIG, SMALL, etc. elements) with Table Headers ("TH") and Table Data ("TD") elements. For exemplary HTML to WML conversion, the following mapping is used for simple tables: HTML <TABLE>content</TABLE> to: WML <CARD>content</CARD> HTML <TBODY>content</TBODY> to: WML <CARD>content</CARD> HTML <CAPTION>content</CAPTION> to: WML <BR ALIGN="CENTER"/>content ALIGN="LEFT"/> HMTL <TR>content</TR> to: WML content HTML <TD>content</TD> to: WML content<TAB/> HTML <TH>content</TH> to: WML content<TAB/> Complex tables includes nested tables and are converted using table unrolling methods. The table unrolling methods indicate table cells in an output format, i.e., enclose table cells within brackets (e.g., "[" and "]") and begins new rows on new lines.
Converting frame elements	Frame elements in the first markup language are converted into markup in the second wireless markup language that preserves the general structure of frames (e.g., from an HTML FRAME ELEMENT, a WML card with links to the HTML frames, and a card with the content of the HTML NOFRAMES element if it exists. The value of the HTML SRC attribute for the FRAME element is mapped to a WML GO element. If it exists, the value of the HTML NAME attribute for the FRAME element is used to identify a link. Otherwise, a link is named "FrameN", where N is a sequential number. For example, the following HTML markup, <FRAME NAME="name" SRC="URL"> is mapped to the WML markup, <A TITLE="name">name<GO URL="URL"/></A>).
Converting link elements	Link elements in the first markup language are converted into link elements in the second wireless markup language. For exemplary HTML to WML coversion, the following mapping is used for links: HTML <A NAME="name" HREF="url">link text </A> to: WML <A TITLE="name">link text<GO URL="url"/><A/>

Table 2.

At Step 32, one or more of the original the non-textual elements are converted from an original non-textual format for the first markup language into a converted non-textual format for the second wireless markup language. In one preferred embodiment of the present invention, non-textual format such as graphical images and image maps are converted from their original format into a format useable with the second wireless markup language. For example, BitMaPed ("BMP") images and Joint Pictures Expert Group ("JPEG") images are converted into a Wireless Bit-MaPped ("WBMP") image format defined by the Wireless Application Protocol, and incorporated herein by reference. BMP and JPEG image format are known to those skilled in the art.

Table 3 illustrates exemplary conversions used to convert non-textual elements for the first markup language into non-textual elements for the second wireless markup language at Step 32. However, more or fewer non-textual conversions could also be used, and the present invention is not limited to the non-textual conversions listed in Table 3.

Non-textual conversion	Description
Converting Images	Images in formats used by the first markup language are converted into a format useable by the second wireless markup language (e.g., JPEG images are converted to a WBMP format required by WAP). Images can also be replaced with links to the original image to improve performance and conserve display space.
Converting client-side image maps	Client-side image maps in the first markup language are converted into markup for the second wireless markup language to allow users to access links mapped to an original image(e.g., for HTML IMG elements or OBJECT elements with USEMAP attributes that refer to MAP elements within an HTML document, a WML CARD element containing links for HTML AREA elements within the HTML MAP that contains a HTML HREF attribute, and the image or object itself is

	created).

Table 3.

Table 3 illustrates only image conversion for non-textual elements. However, the present invention is not limited to image conversions and conversion for other non-textual elements (e.g., audio, video, animation, etc.) can also be used.

In one exemplary preferred embodiment of the present invention, the psuedo-code listing in Table 4 includes exemplary C/C++ data structures and interfaces for conversion functions for textual elements used at Step 30 and non-textual elements at Step 32 of Method 24. In one exemplary preferred embodiment of the present invention, the exemplary C/C++ data structures and interfaces from Table 4 are used with the exemplary psuedo-code from Table 1 to divide an original document at Step 28 and convert original document elements at Steps 30 and 32.

However, the present invention is not limited to the C/C++ data structures or interfaces from Table 4, using the data structures and interfaces from Table 4 with the pre-processing pseudo-code from Table 1. More or fewer elements and element attributes and more or fewer conversion functions can also be used for preferred embodiments of the present invention. In addition, the present invention is not limited to object-oriented programming techniques (e.g., C++), and non-object oriented programming languages (e.g., the C programming language) could also be used for conversion functions.

```
© 1998 by Spyglass, Inc.

// Convert elements and element attributes

// Attribute list (i.e. keep these attributes in a conversion table and delete the rest) table
// table fields:
// HTML tagName, attributeName
```

```
static tagAttrPair attrWhitelist[] = {
        {"A","HREF"},
        {"A","NAME"},
        {"A","TITLE"},
        {"AREA","ALT"},
        {"AREA","HREF"},
        {"BASE","HREF"},
        {"FIELDSET","TITLE"},
        {"FORM","ACTION"},
        {"FORM","METHOD"},
        {"FRAME","NAME"},
        {"FRAME","SRC"},
        {"IMG","ALT"},
        {"IMG","SRC"},
        {"IMG","USEMAP"},
        {"INPUT","MAXLENGTH"},
        {"INPUT","NAME"},
        {"INPUT","SIZE"},
        {"INPUT","TITLE"},
        {"INPUT","TYPE"},
        {"INPUT","VALUE"},
        {"LINK","HREF"},
        {"MAP","NAME"},
        {"OPTGROUP","TITLE"},
        {"OPTION","TITLE"},
        {"SELECT","NAME"},
        {"SELECT","TITLE"},
        {"SELECT", "MULTIPLE"},
        {"TEXTAREA","COLS"},
        {"TEXTAREA","NAME"}
};

// HTML tag translation data table
// table fields:
// tag, tagAction, replacementTag, replStartOnlyFlag, replEndOnlyFlag
static tagTrans tagTransData[] = {
        {"APP", removeWithContent, "", FALSE, FALSE},
        {"APPLET", removeWithContent, "", FALSE, FALSE},
        {"COMMENT", removeWithContent, "", FALSE, FALSE},
        {"DEL", removeWithContent, "", FALSE, FALSE},
        {"PLAINTEXT", removeWithContent, "", FALSE, FALSE},
        {"S", removeWithContent, "", FALSE, FALSE},
        {"SCRIPT", removeWithContent, "", FALSE, FALSE},
        {"STRIKE", removeWithContent, "", FALSE, FALSE},
        {"STYLE", removeWithContent, "", FALSE, FALSE},

        {"ABBR", removeWithoutContent, "", FALSE, FALSE},
        {"ABBREV", removeWithoutContent, "", FALSE, FALSE},
        {"ACRONYM", removeWithoutContent, "", FALSE, FALSE},
        {"AU", removeWithoutContent, "", FALSE, FALSE},
        {"BANNER", removeWithoutContent, "", FALSE, FALSE},
        {"BASEFONT", removeWithoutContent, "", FALSE, FALSE},
        {"BDO", removeWithoutContent, "", FALSE, FALSE},
        {"BGSOUND", removeWithoutContent, "", FALSE, FALSE},
```

```
{"COL", removeWithoutContent, "", FALSE, FALSE},
{"COLGROUP", removeWithoutContent, "", FALSE, FALSE},
{"DIR", removeWithoutContent, "", FALSE, FALSE},
{"DIV", removeWithoutContent, "", FALSE, FALSE},
{"DL", removeWithoutContent, "", FALSE, FALSE},
{"FIG", removeWithoutContent, "", FALSE, FALSE},
{"FN", removeWithoutContent, "", FALSE, FALSE},
{"FONT", removeWithoutContent, "", FALSE, FALSE},
{"INS", removeWithoutContent, "", FALSE, FALSE},
{"ISINDEX", removeWithoutContent, "", FALSE, FALSE},
{"LABEL", removeWithoutContent, "", FALSE, FALSE},
{"LANG", removeWithoutContent, "", FALSE, FALSE},
{"LH", removeWithoutContent, "", FALSE, FALSE},
{"META", removeWithoutContent, "", FALSE, FALSE},
{"MENU", removeWithoutContent, "", FALSE, FALSE},
{"NEXTID", removeWithoutContent, "", FALSE, FALSE},
{"NOTE", removeWithoutContent, "", FALSE, FALSE},
{"OL", removeWithoutContent, "", FALSE, FALSE},
{"OVERLAY", removeWithoutContent, "", FALSE, FALSE},
{"PARAM", removeWithoutContent, "", FALSE, FALSE},
{"PERSON", removeWithoutContent, "", FALSE, FALSE},
{"Q", removeWithoutContent, "", FALSE, FALSE},
{"SOUND", removeWithoutContent, "", FALSE, FALSE},
{"SPAN", removeWithoutContent, "", FALSE, FALSE},
{"TBODY", removeWithoutContent, "", FALSE, FALSE},
{"TFOOT", removeWithoutContent, "", FALSE, FALSE},
{"THEAD", removeWithoutContent, "", FALSE, FALSE},
{"UL", removeWithoutContent, "", FALSE, FALSE},
{"WBR", removeWithoutContent, "", FALSE, FALSE},
{"NOSCRIPT", removeWithoutContent, "", FALSE, FALSE},
{"XMP", removeWithoutContent, "", FALSE, FALSE},

{"ADDRESS", replace, "I", FALSE, FALSE},
{"BLINK", replace, "STRONG", FALSE, FALSE},
{"KBD", replace, "B", FALSE, FALSE},
{"MARQUEE", replace, "STRONG", FALSE, FALSE},
{"SAMP", replace, "I", FALSE, FALSE},
{"SUB", replace, "SMALL", FALSE, FALSE},
{"SUP", replace, "SMALL", FALSE, FALSE},
{"TAB", replace, "TAB/", FALSE, FALSE},
{"VAR", replace, "I", FALSE, FALSE},
{"BLOCKQUOTE", replace, "BR", FALSE, TRUE},
{"BQ", replace, "BR", FALSE, TRUE},
{"CAPTION", replace, "BR", FALSE, TRUE},
{"CENTER", replace, "BR", FALSE, TRUE},
{"CITE", replace, "I", FALSE, FALSE},
{"CODE", replace, "I", FALSE, FALSE},
{"CREDIT", replace, "BR", FALSE, TRUE},
{"DD", replace, "BR", FALSE, TRUE},
{"DFN", replace, "I", FALSE, FALSE},
{"DT", replace, "BR", FALSE, TRUE},
{"HR", replace, "BR", FALSE, TRUE},
{"H1", replace, "BR", FALSE, TRUE},
{"H2", replace, "BR", FALSE, TRUE},
```

```
        {"H3", replace, "BR", FALSE, TRUE},
        {"H4", replace, "BR", FALSE, TRUE},
        {"H5", replace, "BR", FALSE, TRUE},
        {"H6", replace, "BR", FALSE, TRUE},
        {"LEGEND", replace, "BR", FALSE, TRUE},
        {"LI", replace, "BR", FALSE, TRUE},
        {"LISTING", replace, "BR", FALSE, TRUE},
        {"NOBR", replace, "BR", FALSE, TRUE},
        {"PRE", replace, "BR", FALSE, TRUE},
        {"TR", replace, "BR", FALSE, TRUE},
        {"TT", replace, "BR", FALSE, TRUE}
};
```

Table 4.

As is illustrated in Table 4, the first data structure includes a list of element attributes that will be retained during preprocessing in Table 1. Other attributes will be removed. The second data structure determines how the element tags will be converted. As is illustrated in the second data structure, a tag action routine "removeWithoutContent" removes an element tag and saves its content (e.g., corresponding text, etc). A tag routine "removeWithContent" removes an element tag and does not save its content. A tag routine "replace" replaces an element tag in a first markup language with an element in a second markup language (e.g., an HTML element is replaced with a WML element). Two replacement flags, "replStartOnlyFlag" and "replEndonlyFlag" indicate how an element tag is to be replaced. However, the present invention is not limited to the element tag list or element tag conversion data structures and related conversion routines, and other lists, data structures and conversion elements could also be used. In addition, more or fewer elements could also be used for conversion.

In one exemplary preferred embodiment of the present invention, the C/C++ psuedo-code listing in Table 5 includes exemplary routines for conversion functions for non-textual elements used at Step 32 of Method 24. Table 5 illustrates conversions

specifically for converting a JPEG image format to WBMP image format. In Table 5, the variable "data" includes the original image data which is modified by the routine and the variable "metaData" includes HTTP header data. Other similar functions would be used to convert other non-textual elements (e.g., other image formats, audio, video, animation, etc.). However, the present invention is not limited to the routines listed in Table 5 for converting JPEG images into WBMP images, and more or fewer conversion functions for other image formats can also be used.

In another exemplary preferred embodiment of the present invention, JPEG images are converted directly into WBMP images without the intermediate steps of converting the JPEG image into an internal format, converting the internal format into BMP image, and converting the BMP image into a WBMP image. In such an embodiment, a single conversion function is used (e.g., JPEGToWBMP(*data)).

```
© 1998 by Spyglass, Inc.

// Convert JPEG images to WML WBMP

//Convert JPEG format images to WBMP images
//Convert JPEG to an internal format
Convert JPEG_ToInternalFormat(char *data, char *metaData);

//Convert from an internal format to BMP format
//In this routine, data is translated from internal format to BMP format
//metaData is changed as follows:
//Content-Length" is updated
//Content-Type" is change to "image/bmp"
ConvertInternalFormatToBMP(char *data, char *metaData)

//Convert from BMP format to WBMP format
//metaData is changed as follows:
//Content-Length" is updated
//Content-Type" is change to "image/x-wap.wbmp"
ConvertBMP_ToWBMP(char *data, char *metaData)
```

Table 5.

At Step 34, a converted electronic document is created from the converted textual elements and the converted non-textual elements. In one exemplary preferred embodiment of the present invention, the converted electronic document is an electronic document converted with the WAP including WML and protocols from the WAP (e.g., WBMP for images) from an original HMTL document. However, the original electronic document may include other markup languages beside HTML (e.g., cHTML, XML, SGML, VRML, VoxML, etc.), and the converted electronic document may include other wireless markup languages.

In addition, in one exemplary preferred embodiment of the present invention, Step 34 includes querying database 18 for wireless device preferences before the converted electronic document is created. In such an embodiment, the converted textual and non-textual elements may be further modified to meet specific device preferences. For example, a first wireless device 12 may be of a first device type that has a display of 5-lines of 30 characters for textual display. Another wireless device may be of a second device type that has a display with 2-lines of 20-characters for textual display. The converted textual and non-textual elements may be further modified at Step 34 to take into account the wireless device type before creating a final version of a converted electronic document.

At Step 36, the converted electronic document is sent from the second network device 14 to the first wireless device 12 in response to the request for the original electronic document. In one preferred embodiment of the present invention, a WAP document including WML and other WAP protocols (e.g., WBMP for images) is sent from the second network device 14 to the first wireless device 12 in response to the

request for an original HTML document. However, other original electronic document and converted document types could also be used and the present invention is not limited to HTML-to-WAP conversions.

In one exemplary preferred embodiment of the present invention, Step 36 includes determining if the first wireless device has enough memory to store the whole converted document (e.g., from the device type in the request header). If not, the converted document is split into multiple sub-documents that are sent one, or a few at a time. For example, a WML deck with N-number of cards could be split into two WML decks, the first deck, including N-X-number of cards, and the second deck including the remaining WML cards. Method 26 allows a wireless device to request virtually any original electric document from the world-wide-web and receive a converted document suitable for display on the wireless device.

Division of an original electronic document

As was discussed above, there are three separate methods used to divide or divide and convert an original electronic document into the textual elements and non-textual elements from the first markup language to be converted into textual elements and non-textual elements in a second wireless markup language at Step 28 of Method 24 (FIG. 2). The three methods include logical division of electronic document elements, categorized division of electronic document elements, and targeted conversion of electronic document elements. However, more or fewer division methods can also be used and the present invention is not limited to three methods described.

Logical division of electronic document elements

This division method preserves as much of the structure of the original electronic document as possible when it is converted into a converted electronic document. The converted document is displayed in a same logical sequence as the original electronic document.

FIG. 3 is a flow diagram illustrating a Method 40 for logical division of electronic document elements. At Step 42, a first set of document elements in a first markup language is selected for dividing an original electronic document into multiple logical sub-documents in a second wireless markup language (e.g., see Table 6). The first set of document elements includes textual elements and non-textual elements from the first markup language. At Step 44, the original electronic document is parsed for document elements from the first set of elements (e.g., see Table 1). At Step 46, the original electronic document is divided into multiple of sub-documents for a second wireless markup language using parsed document elements from the first set of elements. The multiple of sub-documents for the second wireless markup language maintain a logical sequencing associated with the parsed document elements from the original electronic document.

In one exemplary preferred embodiment of the present invention, the first markup language is HTML and the second wireless markup language is WML. However, other markup languages can also be used and the present invention is not limited to HTML and WML, another other markup languages could also be used (e.g., cHTML, SGML, XML, VRML, VoxML, etc.). In such an embodiment, at Step 42 a first set of elements in HTML is selected for dividing an original electronic document into logical cards in

WML. The first set of elements includes textual elements and non-textual elements from HTML.

Table 6 illustrates exemplary document elements selected for HTML that are used for an exemplary first set of elements. However, the present invention is not limited to the exemplary documents elements illustrated in Table 6, and more or fewer document elements can also be used. In addition, the present invention is not limited to HTML elements and document elements from other markup languages (e.g., cHTML, SGML, XML, VRML, VoxML, etc.) can also be used.

WML card breaks are inserted before the following HTML elements:
TABLE elements
IMG elements that contain a USEMAP attribute
FIELDSET elements
IFRAME elements
NOSCRIPT elements
TBODY elements
NOFRAMES elements
FRAME elements

Table 6.

At Step 44, the original electronic document is parsed for document elements from the first set of elements such as those illustrated in Table 6. At Step 46, the original electronic document is divided into multiple WML cards using the parsed document elements. The multiple WML cards maintain a logical sequencing associated with the parsed document elements from the original electronic document.

FIG. 4 is a block diagram 48 visually illustrating the Method 40 for logical division of electronic document elements. An original HTML electronic document 50 is divided into multiple WML cards 52, 54, 56 that create a WML deck 58. The individual WML cards 52, 54, 56 are linked together with WML links 60 and 60'. The logical sequencing associated with the HTML document 50 is maintained in the WML deck 58.

For example, using the exemplary HTML elements illustrated in Table 6, a HTML TABLE element is the first element parsed from the HTML document 50. A first WML card 52 is created to include the table data associated with the HTML TABLE element. The second HTML element parsed is a HTML IMG element with a USEMAP attribute. The second WML card 54 includes an image associated with the HTML IMG element with the USEMAP attribute. The third HTML element parsed is a HTML FRAME element. The third WML card 56 includes the frame data associated with the HTML FRAME element. Thus, using Method 40 to create the WML deck 60, preserves the logical sequencing associated with document elements from the original HTML document 52. Method 40 (FIG. 3) can be used at Step 28 of Method 24 (FIG. 2) to divide an original electronic document into multiple document elements.

Categorized division of electronic document elements

This method divides an original electronic document into general categories of document elements used to create an original electronic document (e.g., text, links, images, tables, etc.). The categories are displayed in a list that allows navigation to the individual categories.

FIG. 5 is a flow diagram illustrating a Method 62 for categorized division of electronic document elements. At Step 64, a first set of categories for a first markup language for dividing an original electronic document into sub-documents in a second wireless markup language is selected. An individual category in the first set of categories is associated with one or more document elements from the first markup language. At Step 66, a structured model is created for the original electronic document (e.g., from the token list created in Table 1). At Step 68, a loop is entered to parse the structured model

for one or more document elements from the original electronic document that are included in the first set of categories. At Step 70, an individual category is added to a first list associated with a first sub-document for a second wireless markup language. At Step 72, parsed one or more document elements are added to a next list associated with the individual category on a next sub-document for the second wireless markup language. At Step 74, an association between the first sub-document and the next sub-document is created. The association includes state information. Steps 68, 70, 72 and 74 are repeated until the end of the structured model for the original electronic document to divide the original electronic document into a plurality of sub-documents for the second wireless markup language. In addition, Step 72 can be repeated to create further sub-documents associated with the next sub-document.

In one exemplary preferred embodiment of the present invention, the first markup language is HTML and the second wireless markup language is WML. However, other markup languages can also be used and the present invention is not limited to HTML and WML, another other markup languages could also be used (e.g., cHTML, SGML, XML, VRML, VoxML, etc.). In such an embodiment, at Step 64 a first set of categories for dividing an original HTML electronic document into WML cards in a WML card deck is selected. HTML (e.g. the HTML IMG element is associated with an image category, the HTML TABLE element is associated with a table category, etc.). An individual category in the first set of categories is associated with one or more document elements from the original HTML electronic document.

Table 7 illustrates exemplary categories selected for HTML elements used for an exemplary first set of categories. However, the present invention is not limited to the

exemplary categories illustrated in Table 7, and more or fewer categories can also be used. In addition, the present invention is not limited to HTML categories and categories from other markup languages (e.g., cHTML, SGML, XML, VRML, VoxML, etc.) can also be used.

WML cards are created from HTML elements associated with following categories:
Links
Text
Tables
Input Fields
Page Sections
Links from Images
Images
Navigation

Table 7.

At Step 66, a structured model is created for the original electronic document. In one exemplary preferred embodiment of the present invention, a Document Object Model ("DOM") is created for the original electronic document at Step 68. The DOM defines a logical structure of hypertext electronic documents and illustrates how hypertext electronic documents are accessed and manipulated. However, other structured models can also be used and the present invention is not limited to the DOM. Table 8 illustrates exemplary routines for creating a DOM at Step 66. However, more or fewer routines could also be used.

Conversion Operation	Description
DOM_CREATE	Creates a Document Object Model ("DOM") from an electronic document.
DOM_EVAL	Evaluates one or more expressions that select one or more hypertext elements from a DOM.

Table 8.

FIG. 6 is a block diagram illustrating a portion of an exemplary Document Object Model ("DOM") 76 for HTML. For more information on the DOM for HTML see the World-Wide-Web Consortium ("W3C") Document Object Model, incorporated herein by

reference. The W3C DOM can be found on the World-Wide-Web at the URL "http://www.w3.org." Dynamic HTML by Microsoft was also one immediate ancestor of the W3C Document Object Model. More information on Dynamic HMTL ("DHTML") can be found at the URL "http://www.microsoft.com." However, the present invention is not limited to a DOM for HTML, or DHTML, and a DOM can also be created for other markup languages (e.g., cHTML, SGML, XML, VRML, VoxML, etc.).

Returning to FIG. 6, the exemplary DOM 76 for HTML includes a document type of HTML 78, an HTML element HEAD 80, a HTML HEAD sub-element TITLE 82, and a sub-element text 84 for the sub-element TITLE 82. The HTML element BODY 86 includes a sub-element HEADING 88 and text 90. The BODY 86 sub-element PARAGRAPH 92 includes text 94. The BODY 86 sub-element TABLE 96 includes sub-elements ROWS 98, 100. Sub-element ROW 98 includes sub-element CELL 102. Sub-element CELL 102 includes text 104. Additional HTML BODY 86 sub-elements are included in the DOM 76 beginning at 106. Other HTML elements, sub-elements attributes and text are also added to the DOM 76 for HTML, but are not illustrated in FIG. 6.

Returning to FIG. 5 at Step 68, a loop is entered to parse the DOM 76 for one or more document elements (e.g., HTML TABLE element 96 (FIG. 6)) from the original HTML electronic document that are included in categories from the first set of categories illustrated in Table 7.

At Step 70, an individual category (e.g., table) is added to a first list associated with a first WML card. At Step 72, the parsed one or more document elements (e.g.,

HTML TABLE text) are added to a next list associated with the individual category (e.g., table) on a next WML card. At Step 74, a WML link is created between the first WML card and the next WML card. The WML link includes state information that allows other WML cards to be accessed. Steps 68, 70, 72 and 74 are repeated until the end of the DOM for the original electronic document. A deck of WML cards is created.

In addition, Step 72 can be repeated to create further sub-documents associated with the next sub-document. For example, the first WML card may contain a first list including a category for links. A next WML card may contain a second list including a list of links from the HTML document. WML cards next-sub+1 through next-sub+N may include content associated with links from the second list of links.

FIG. 7 is a block diagram 110 visually illustrating the Method 62 for categorized division of electronic documents of FIG. 5. A first WML card 112 includes a first list of categories (e.g., text, links, and tables). A next WML card 114 includes text. A next+1 WML card 116 includes a second list of links. A next+2 WML card 118 includes a table data. WML links (i.e., associations), 120, 120', 120" are created between the first WML card 112 and the next WML cards 114, 116, and 118. The WML cards 112, 114, 116 and 118 form a WML card deck 122. Method 62 (FIG. 5) can also be used at Step 28 of Method 24 (FIG. 2) to divide an original electronic document into multiple electronic documents.

In one exemplary preferred embodiment of the present invention, a first WML card in a WML deck created with Method 62 includes a first list with general category choices (e.g., those illustrated in Table 7). This first list is a scrollable list. The first seven list items (e.g., links, text, tables, input fields, page selections, links from images,

and images) are WML links to information from the original HTML electronic document. The WML links contain state information for a category. The eighth item allows further navigation among the first seven items.

The content converter application 16 populates as many WML cards as possible. For wireless device with enough memory to display one deck of WML cards, the first WML card is linked to other WML cards in the same deck. If the wireless device does not have enough memory to display a single deck of WML cards, the content converter application 16 uses state information and creates another deck and links the decks to present the HTML information. For example, if the original HTML electronic document is at the URL "http://www.abc.com," the link represented by a list item in a first list on the first WML card for the category links is a virtual URL "http://www.abc.com_spyg_(SPYGA-HTML2WML-STATE=L1)," where, (_spyg_(SPYGA-HTML2WML-STATE=L1), is the state information. A wireless device selecting the "http://www.abc.com_spyg_(SPYGA-HTML2WML-STATE=L1)" virtual URL is requesting that the content converter application 16 display a first WML deck of cards. A next WML deck of cards would include state information, (_spyg_(SPYGA-HTML2WML-STATE=L2), where the L2 indicates a second state after the first state L1. The content converter application 16 decodes the virtual URLS.

The wireless device can also have wireless device preferences retrieved from the database 18 (FIG. 1) to limit the amount of lines displayed on a wireless device display. The state information is then used in a similar manner based on the wireless device preferences.

Table 9 illustrates exemplary WML actions for the category choices illustrated in Table 6. However, other actions could also be provided for the category choices, and the present invention is not limited to the actions listed in Table 9.

HTML Category	WML Action
Links	Provide a list of links from an original HTML document,
Text	Provide text from an original HTML document without the HTML markup tags.
Tables	Provide the table data from an original HTML table.
Input Fields	Provide an input element used to represent HTML input fields in HTML forms.
Page Selections	Provide a selection list of links to frames.
Links from Images	Provides a selection list of links as defined by a client-side image map.
Images	Provides a selection list of links to the images for wireless devices with graphics capabilities.
Navigation	Provide an entry WML card that takes as input a state index. With the entry card and state index, provide navigation to other WML cards.

Table 9.

The actual amount of information displayed on a wireless display is dependent upon the resolution of the wireless display (e.g., 5-lines of 18-characters). The amount of information (e.g., WML cards) sent to a wireless device is dependent on the amount of memory on the wireless device. State information is used to connect multiple decks of WML cards.

Targeted conversion of electronic document elements

This method allows expressions to be created that specify which document elements to extract from an original electronic document. The expressions allow specific document elements at specific locations to be extracted from an original electronic document (e.g., a specific stock quote from a list of stock quotes, or a specific score from a list of sports score).

FIG. 8 is a flow diagram illustrating a Method 124 for targeted conversions of electronic document elements. At Step 126, a set of extraction expressions is created to extract one or more document elements from an original electronic document. The document elements include textual and non-textual elements. At Step 128, the set of extraction expressions is applied to the original electronic document to extract one or more document elements. At Step 130, the original electronic document is divided into a plurality of sub-documents for a second wireless markup language using the extracted document elements.

In one exemplary preferred embodiment of the present invention, the content converter application 16 converts the extracted HTML elements into WML elements either by dynamically generating the WML based on characteristics of the extracted HTML elements, or by writing the extracted HTML elements to a WML template. The WML template is a pre-determined set of WML cards in one or more WML decks. In one exemplary preferred embodiment of the present invention, the extraction expressions are applied at Step 128 to a DOM 76 (FIG. 6) created from the original electronic document. In another exemplary preferred embodiment of the present invention, the extraction expressions are applied to list of document elements created from the original electronic document.

For example, suppose a user is a baseball fan and desires to see only a box score for his/her favorite team on a day-to-day basis. In one preferred embodiment of the present invention, a user can select a desired box score from a sports page on an electronic document server on the World-Wide-Web (e.g., from a newspaper sports page)

for daily display on a wireless device. An exemplary baseball box score format from an exemplary sports page is illustrated in Table 10.

Final: Fri. April 17, 1998												
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**R**	**H**	**E**
Chicago-C	0	0	0	0	0	0	0	2	0	2	5	0
New York –M	0	1	0	0	0	0	0	0	0	1	3	1

Table 10.

One exemplary preferred embodiment of the present invention allows a user the capability not only to select one or more desired hypertext elements to use on a wireless device, but also the capability to alter the content to a format suitable for the user. Suppose the user only desires to determine if his/her favorite team has won or lost a game and has only enough display space on a wireless device to display a summary of a game. Table 10 illustrates a desired box score format suitable for a wireless device. Table 11 illustrates an abbreviated summary of the original baseball box score suitable for display on a wireless device. In one embodiment of the present invention, the abbreviated summary of the original baseball box score suitable for display on a wireless device would be sent to the wireless device in WML after extracting the appropriate information from HTML.

Final: Fri. April 17, 1998			
	Runs	**Hits**	**Errors**
Chicago-C	2	5	0
New York-M	1	3	1

Table 11.

In one preferred embodiment of the present invention, the extraction expressions and operations are from a data mining conversion language. However, other extraction expressions can also be used and the present invention is not limited to extraction expression from a data mining language. The data mining conversion language is a set of

object-oriented methods that can be used to identify one or more hypertext elements in a hypertext electronic document. They can also be used to obtain text and change attribute values for the hypertext elements. However, non-object-oriented methods can also be used.

Table 12 illustrates exemplary data mining object-oriented methods for an exemplary data mining conversion language. However, more or fewer object-oriented methods could also be used in the exemplary data mining conversion language. The object-oriented methods provide component names used to create data mining expressions.

Hypertext Element Desired	Method
A hypertext electronic document.	document
All hypertext elements in a hypertext electronic document.	all
The n-th hypertext element in a list.	item (n-1) (Since zero based indexes are used, n-1 returns the nth item).
Hypertext elements with a specified tag.	tags ("tag")
Hypertext elements that contain an attribute name/value pair.	attributeMatch(name=value[,name=value])
Search for elements with the specified text.	grep("text")

Table 12.

Table 13 illustrates exemplary object-oriented methods used for sub-elements for an exemplary data mining conversion language. However, more or fewer object-oriented methods could also be used.

Element	Sub-Element	Method
table	One or all rows of a table.	rows(m) or rows
table row	One or all cells in a table row.	cells(m) or cells
image map	One or all areas defined in an image map.	areas(m) or areas
form	One or all elements in a form.	elements(m) or elements
form select	One or all options in a select list.	options(m) or options
other (e.g., lists and frames)	One or all sub-elements of an element.	children(m) or children

Table 13.

The data mining conversion language also includes, for example, methods for getting the text of a hypertext element, getting a value of an attribute for an element, and for searching for text of an element. Exemplary object-oriented methods for getting the text of an element are illustrated with exemplary data mining expressions in Table 14.

Task	Sample data mining expression
Get the text of a bullet list item.	document.all.tags("BL").item(n).children(m).text
Get the text from a table cell.	document.all.tags("TABLE").item(n).rows(rownum).cells(colnum).text
Get the text from all cells in a row. The text is concatenated into one string.	document.all.tags("TABLE").item(n).rows(rownum).text
Get the text from all cells in a column. The text is concatenated into one string.	document.all.tags("TABLE").item(n).rows.(rownum).cells(cellnum).text
Get options from a SELECT element. Text is concatenated into one string.	document.all.tags("FORM").item(n).elements.tags("SELECT").item(n).options.text

Table 14.

As an example, to get the text of a table list item-n with the exemplary data mining expression "document.all.tags("TABLE").item(n).children(m).text," the object-oriented method "document" (Table 12) is combined with the methods "all" (Table 12), "tags" (Table 12), for the HTML table list tag "TABLE," "item" (Table 12), "children" (Table 13) and "text" (Table 14). The data mining language expression (e.g., Table 14) obtains text from all HTML bullet list tags, including nested tags in a HTML document. Other data mining expressions in Table 14 are created in a similar manner to obtain text from HTML tables and forms.

Object-oriented methods for getting a value for an attribute of hypertext element using data mining expressions are illustrated with exemplary data mining expressions in Table 15. The exemplary data mining expressions combine object-oriented methods from Table 12 and Table 13.

Task	Sample data mining expression
Get the color of visited hyperlinks	Document.all.tags("BODY").getAttribute("vlink")
Get the URL of an anchor	Document.all.tags("A").item(n).getAttribute("href")

Get the URL associated with an area in an image map	Document.all.tags.("MAP").areas(areanum).getAttirbute("href")
Get the URL of a form	Document.all.tags.("FORM").item(n).getAttribute("action")

Table 15.

One object-oriented method for searching for text of a hypertext element using data mining expressions are illustrated with an exemplary data mining expression in Table 16. The text search method searches for a specified string and returns all elements that contain it. As a result, duplicate text information may be returned if the text search method is not used at the bottom of the element hierarchy in the DOM 76 (FIG. 6).

Task	Sample data mining expression
Search in a frame for an element containing the text "Contents."	Document.all.tags("FRAME").grep("CONTENTS")

Table 16.

The exemplary data missing expressions can be used to extract information for display on wireless device. Exemplary extraction expressions may be stored in database 18. (FIG. 1). FIG. 9 is a block diagram 132 visually illustrating the Method 124 for targeted conversion of electronic document elements of FIG. 8. An original HTML electronic document 134 includes a table 136 including a sports score (e.g., Table 10). An extraction expression 138 is applied to the table 136 to create a WML card 140 in a WML deck 142 that includes a subset of data from the original table 136 (e.g., Table 11). Method 128 (FIG. 8) can also be used at Step 28 of Method 24 (FIG. 2) to divide an original electronic document into multiple document elements.

Illustrative uses of preferred embodiments of the present invention

FIGS. 10 and 11 visually illustrate use of preferred embodiments of the present invention. However, FIGS. 10 and 11 are exemplary only, and the present invention is not limited to the displays illustrated. FIG. 10 is screen display 142 illustrating visual

output from an exemplary HTML electronic document from the World-Wide-Web. The exemplary electronic document includes target text 144 in a "Latest News" section of the screen display 142. Table 17 illustrates a portion of exemplary HTML used to display the screen display 142 from FIG. 10. In the exemplary HTML illustrated in Table 17, the text 144 from the "Latest News" section is stored in an HTML table. However, target text 144 could also be stored with other HTML elements, and Table 17 is exemplary only.

```
© 1998 by Spyglass, Inc.

<HTML>
<!—Copyright 1998 Spyglass -->
<HEAD>
<TITLE>Spyglass, Inc. I Making Devices Work with the Web</TITLE>
        <meta name="Description" content="Spyglass is the leading provider of Internet expertise,
software and services for making devices work with the Web. Spyglass solutions allow our
customers to unleash the power of the Internet.">
        <meta name="Keywords" content="Spyglass Spyglass, SurfWatch, Professional Services,
NASDAQ:SPYG, SPYG,
        Device Mail, DeviceMail, ThinGUI Library, Device Mosaic, Remote Mosaic, MicroServer,
Prism, embedded server, NetHopper, Mobile Forms Database for Windows CE,
        strategic internet consulting, product service design, implementation, planning, custom
application development, diagnostics, content services, porting, integration, QA, Quality
Assurance, testing, training, installation, support services, internet software, consulting, software
engineering, solutions provider, custom product development,
        RTOS, real time Real-time operating systems, embedded systems, content conversion,
web device, browsers, internet appliance, internet television, wirelesss data services, email, e-
mail, content filtering, non-PC devices, scalable embedded browsers servers, Java, p-Java,
personal Java, Windows CE development ">
        <meta http-equiv="Window-target" Content="_top">
        <STYLE TYPE="text/css">
         <!--
         .SIDEBAR {font-size: 11px; text-decoration:none; font-weight:medium; font-family:
verdana,          arial, helvetica, sans-serif;}
         -->
        </STYLE>
</head>
<table border=0 cellpadding=0 cellspacing=0 width=580>
<tr> <! row 1>

        <td width=200 valign=top>
        <img width=200 height=45 src="/images/logo.gif"><br>
        </td>
        <td width=380 valign=top rowspan=2>
        <img border=0 width=380 height=24 src="/images/tag.gif"><br>
```

```
        <img usemap="#mainnavmap" border=0 width=380 height=51
src="/images/navhome.gif"><br>
        </td>
</tr>
<tr> <! row 2>
        <td width=200 valign=top>
        <img width=200 height=60 src="/images/leftcurvehome.gif"><br>
        </td>
</tr>
<tr> <! row 3>
        <td width=200 valign=top>
        <table cellspacing=0 cellpadding=0 border=0 width=200>
        <tr> <! row 4>
         <td width=10 valign=top bgcolor="#003366"> <! item 1>
         <img width=10 height=1 src="images/clear.gif"><br>
         </td>
         <td width=10 valign=top> <! item 2>
         <img width=10 height=1 src="images/clear.gif"><br>
         </td>
         <td width=145 valign=top align=right> <! item 3>
         <img width=145 height=25 src="images/latestnews.gif"><br>
<!-- Latest News items here - limit 3 --><DIV ID="news">
         <font face="Verdana, Arial, Helvetica" size="2">
<! cell 1>
         <a href="/newsflash/releases/113098swwin98.html" class=SIDEBAR>Afga To Partner
with Spyglass on Advanced TV Font Rendering <IMG SRC="images/Arrow_rt2.gif" WIDTH=9
HEIGHT=9 BORDER=0 hspace=3 align=middle ALT="Go"></a><br><br>
         <a href="/newsflash/releases/111998swpccomputing.html" class=SIDEBAR>Spyglass
Ships New Version of Device Mosaic<IMG SRC="images/Arrow_rt2.gif" WIDTH=9 HEIGHT=9
BORDER=0 hspace=3 align=middle ALT="Go"></a><br><br>
         <a href="/newsflash/releases/111698wince.html" class=SIDEBAR>Device Mail 3.1
Embedded Email Client Announced<IMG SRC="images/Arrow_rt2.gif" WIDTH=9 HEIGHT=9
BORDER=0 hspace=3 align=middle ALT="Go"></a><br><br>
         </font>
</DIV>
<!-- end Latest News items -->
         </td>
         <td width=35 valign=top>
         <img width=35 height=1 src="images/clear.gif"><br>
         </td>
        </tr>
        <tr>
         <td width=200 valign=top colspan=4>
         <img width=200 height=35 src="/images/bottomnavhome.gif"><br>
         <img width=165 height=128 src="/images/bottombunhome4.gif" border="0"
usemap="#utilitymap"><br>
         </td>
        </tr>
        </table>
...

</html>
```

Table 17.

FIG. 11 is a block diagram 146 illustrating exemplary converted output from the screen display of FIG. 10 on an exemplary first wireless device 12. The text 144 from the "Latest News" section of the screen display 142 (FIG. 10) is displayed on a wireless display 148 on the exemplary first wireless device 12 (FIG. 11). The exemplary electronic document displayed in FIG. 10 is the exemplary HTML document from Table 17. The exemplary converted output displayed on the wireless display 148 is a WML card from a WML document. The HTML document used to create the screen display 142 from FIG. 10 is converted with the one or more of the methods and the system of preferred embodiments of the present invention and displayed on the wireless display 148 as a WML card on first wireless device 12. Other text from the "Latest News" section of the screen display 142 may be displayed with additional WML cards.

As an example, a user desires to see text 144 from the "Latest News" section of the screen display 142 (FIG. 10). The text 144 (FIG. 10) from the "Latest News" section is part of a table, and a table element was from a first set of selected document elements from a first original electronic document. Method 24 (FIG. 2) could be used with Method 40 (FIG. 3) at Step 28 of Method 24 to produce the output on the wireless display 148 of FIG. 11. In such an embodiment, the output on the wireless display 148 of FIG. 11 may be the first WML card 52 (FIG. 4) displayed including text from the original table since the HTML TABLE element appeared a first element the first set of selected document elements (e.g., Table 6). Thus, a logical sequencing associated with the original HTML document (e.g., Table 17) is maintained.

As another example, a user desires to see text 144 from the "Latest News" section of the screen display 142 (FIG. 10). Since the text 144 (FIG. 10) from the "Latest News"

section is part of a table, Method 24 (FIG. 3) could be used with Method 62 (FIG. 5) at Step 28 of Method 24 to produce the output on the wireless display 148 of FIG. 11. A DOM would be created and multiple WML cards are created, using the DOM including a WML card for tables (Table 6). In such an embodiment, the output on the wireless display 148 of FIG. 11 may be the next+2 WML card 118 (FIG. 7) for tables, after a user has already selected the item "3. TABLES" from the first WML card 112 (FIG. 7).

As yet another example, a user desires to see text 144 from a first item from the "Latest News" section of the screen display 142 (FIG. 10). Since the text 144 (FIG. 10) from the "Latest News" section is part of a table, Method 24 (FIG. 2) could be used with Method 124 (FIG. 8) at Step 28 of Method 24 to produce the output on the wireless display 148 of FIG. 11. An appropriate extraction expression 138 (FIG. 9) (e.g., document.all.tags.("TABLE").item(3).rows(4).cells(1).text) would be created at Step 126 of Method 124 (FIG. 8) to extract text from the first item from the "Latest News" section of the screen display 142 from the original table (e.g., Table 17). The first item would appear on the first WML card 140 (FIG. 9). Similar expressions could also be created to extract other items (e.g., 2nd or 3rd items) from the "Latest News" table. Such items would be displayed on other WML cards in the WML card deck 142 (FIG. 9).

Preferred embodiments of the present invention were illustrated with three examples for text from an exemplary table. However, the output on the wireless display 148 of FIG. 11 could also be produced using preferred embodiments of the present invention for other than table data, and the present invention is not limited to displaying table data.

In exemplary preferred embodiments of the present invention, the methods and system are used to allow a wireless device to request an original electronic document (e.g., a Hyper Text Markup Language document) and receive a converted electronic document (e.g., a Wireless Markup Language document) suitable for display on the wireless device. The converted electronic document suitable for display on the wireless device may provide increased satisfaction for a user of a wireless device. Preferred embodiment of the present invention are described with respect to wireless devices. However, the present invention is not limited to wireless devices, and other network devices, that are capable of using the WAP or other wireless markup languages, including those network devices connected to a network with a wire or cable, could also be used to practice the present invention.

In view of the wide variety of embodiments to which the principles of the present invention can be applied, it should be understood that the illustrated embodiments are exemplary only, and should not be taken as limiting the scope of the present invention. For example, the steps of the flow diagrams may be taken in sequences other than those described, and more or fewer elements may be used in the block diagrams.

The claims should not be read as limited to the described order or elements unless stated to that effect. Therefore, all embodiments that come within the scope of the following claims and equivalents thereto are claimed as the invention.

WE CLAIM:

1. In a network with a plurality of network devices, a method of content conversion, comprising the following steps:

receiving a request for an original electronic document on a second network device on a first network, from a first wireless device on the first network, wherein the request includes a device type for the first wireless device;

dividing the original electronic document into a plurality of document elements, including textual elements and non-textual elements for a first markup language used to create the original electronic document, for converting the first markup language into a second wireless markup language;

converting one or more of the original textual elements from the original electronic document in the first markup language into converted textual elements for the second wireless markup language;

converting one or more of the non-textual elements from an original non-textual format for the first markup language into a converted non-textual format for the second wireless markup language;

creating a converted electronic document from the converted textual elements and the converted non-textual elements based on the device type for the first wireless device from the request; and

sending the converted electronic document from the second network device to the first wireless device in response to the request for the original electronic document.

2. A computer readable medium having stored therein instructions for causing a central processing unit to execute the method of Claim 1.

3. The method of Claim 1 wherein the first wireless device is a wireless telephone and the second network device is a Wireless Application Protocol gateway.

4. The method of Claim 1 wherein the first markup language is Hyper Text Markup Language and the second wireless markup language is Wireless Markup Language.

5. The method of Claim 1 wherein the step of dividing the original electronic document into a plurality of document elements includes dividing the original electronic document into a plurality of sub-document elements for a second wireless markup language, wherein the plurality of sub-document elements for the second wireless markup language maintain a logical sequencing associated with the plurality of document elements from the original electronic document.

6. The method of Claim 1 wherein step of dividing the original electronic document into a plurality of document elements includes dividing the original electronic document into a plurality of documents elements using a Document Object Model.

7. The method of Claim 1 wherein the step of dividing the original electronic document into a plurality of document elements includes dividing the original electronic

document into a plurality of sub-documents for a second wireless markup language using a set of document element categories for the first markup language.

8. The method of Claim 1 wherein the step of dividing the original electronic document into a plurality of document elements includes dividing the original electronic document into a plurality of sub-documents for a second wireless markup language using extraction expressions from a data mining language.

9. The method of Claim 1 wherein the step of converting one or more of the non-textual elements from an original non-textual format for the first markup language into a converted non-textual format for the second wireless markup language includes converting original image formats in the original electronic document into a Wireless Application Protocol image format for Wireless Markup Language.

10. The method of Claim 1 wherein the step of creating a converted electronic document from the converted textual elements and the converted non-textual elements includes creating a Wireless Markup Language electronic document with a plurality of cards comprising a Wireless Markup Language deck of cards for display on the first wireless device.

11. The method of Claim 1 further comprising:

sending the request for the original electronic document from the second network device to a third network device on a second network to retrieve the original electronic document; and

receiving the original electronic document on the second network device on the first network from the third network device on the second network.

12. The method of Claim 1 wherein the step of creating a converted electronic document from the converted textual elements and the converted non-textual elements includes:

consulting a database associated with the second network device to determine device preferences for the first wireless network device; and

modifying the converted textual elements and the converted non-textual elements, based on the device preferences for the first wireless network device.

13. The method of Claim 1 wherein the request is a Hyper Text Transfer Protocol request.

14. The method of Claim 1 wherein the step of converting one or more of the original textural elements includes converting at least one but less than all of the original textural elements.

15. The method of Claim 1, wherein the step of converting one or more of the original non-textural elements includes converting at least one but less than all of the original non-textural elements.

16. The method of claim 1 wherein the step of sending the converted document includes determining if the first wireless device has enough memory to store for the whole converted document, and if not, splitting the converted electronic document into a plurality of sub-documents; and sending the plurality of sub-documents instead of the whole converted document.

17. In a network with a plurality of network devices, a method of electronic document division, comprising the following steps:

selecting a first set of document elements in a first markup language for dividing an original electronic document into logical sub-documents in a second wireless markup language, wherein the first set of document elements includes textual elements and non-textual elements from the first markup language;

parsing the original electronic document for document elements from the first set of elements;

dividing the original electronic document into a plurality of sub-documents for a second wireless markup language using parsed document elements from the first set of elements, wherein the plurality of sub-documents for the second wireless markup language maintain a logical sequencing associated with the parsed document elements from the original electronic document.

18. A computer readable medium having stored therein instructions for causing a central processing unit to execute the method of Claim 17.

19. The method of Claim 17 wherein the first markup language includes Hyper Text Markup Language and the second wireless markup language includes Wireless Markup Language.

20. The method of Claim 17 wherein the plurality of sub-documents for the second wireless markup language includes a plurality of Wireless Markup Language cards.

21. The method of Claim 17 wherein the first set of elements in a first markup language is any of an IMG and OBJECT elements with a USEMAP attribute, NOFRAMES element, FRAME element, FIELDSET element, IFRAME element, NOSCRIPT element, TBODY element, or a TABLE element, from Hyper Text Markup Language.

22. In a network with a plurality of network devices, a method of electronic document division, comprising the following steps:

(a) selecting a first set of categories for a first markup language for dividing an original electronic document into a plurality of sub-documents in a second wireless markup language, wherein an individual category in the first set of categories is associated with one or more document elements from the first markup language;

(b) creating a structured model for the original electronic document;

(c) parsing the structured model for one or more document elements from the original electronic document that are included in the first set of categories;

(d) adding an individual category to a first list associated with a first sub-document for a second wireless markup language;

(e) adding parsed one or more document elements to a next list associated with the individual category on a next sub-document for the second wireless markup language;

(f) creating an association between the first sub-document and the next sub-document, wherein the association includes state information; and

(g) repeating steps (c) through (f) until the end of the structured model for the original electronic document to divide the original electronic document into a plurality of sub-documents for the second wireless markup language.

23. A computer readable medium having stored therein instructions for causing a central processing unit to execute the method of Claim 22.

24. The method of Claim 22 wherein the first set of categories for a first markup language includes a category for any of links, text, tables input fields, page selections, links from images, images or navigation.

25. The method of Claim 22 wherein the first markup language is Hyper Text Markup Language and the second wireless markup language is Wireless Markup Language.

26. The method of Claim 22 wherein step (b) creating a structured model for the original electronic document includes creating a document object model for the original electronic document.

27. The method of Claim 22 wherein step (d) adding an individual category to a first list associated with a first sub-document for a second wireless markup language includes adding an individual category to a first list associated with the a first card for Wireless Markup Language.

28. The method of Claim 22 wherein step (e) adding the parsed one or more document elements to a next list associated with the individual category on a next sub-document for the second wireless markup language includes adding parsed one or more document elements to a next list associated with the individual category on a next card for Wireless Markup Language.

29. The method of Claim 22 wherein step (f) creating an association between the first sub-document and the next sub-document includes creating a Wireless Markup Language link between a first card and a next card in a Wireless Markup Language deck of cards.

30. In a network with a plurality of network devices, a method of electronic document division, comprising the following steps:

creating a set of extraction expressions to extract one or more document elements from an original electronic document;

applying the set of extraction expressions to the original electronic document to extract one or more document elements wherein the one or more document elements include textual and non-textual document element; and

dividing the original electronic document into a plurality of sub-documents for a second wireless markup language using the extracted document elements.

31. A computer readable medium having stored therein instructions for causing a central processing unit to execute the method of Claim 30.

32. The method of Claim 30 wherein the step of applying the set of extraction expressions to the original electronic document includes applying the set of extraction expressions to a document object model constructed for the original electronic document.

33. The method of Claim 31 wherein the set of extraction expression are expressions from a data mining language.

34. A system for content conversion, comprising in combination:

a wireless gateway for accepting requests for original electronic documents from a wireless device;

a content converter application for converting an original electronic document in a first markup language into a second converted document in a second wireless markup language suitable for display on a wireless device; and

a database for storing device preferences for a plurality of wireless device types, wherein the device preferences are used to further convert an original electronic document in a first markup language into a second converted document in a second wireless markup language suitable for display on a specific type of wireless device.

35. Software for causing a central processing unit to execute the method of claim 1.

36. Software for causing a central processing unit to execute the method of claim 17.

37. Software for causing a central processing unit to execute the method of claim 22.

38. Software for causing a central processing unit to execute the method of claim 30.

39. A method of content conversion substantially as herein described and with reference to the accompanying drawings.

40. A method of electronic document division substantially as herein described and with reference to the accompanying drawings.

41. A system for content conversion substantially as herein described and with reference to the accompanying drawings.

42. A computer readable medium substantially as herein described and with reference to the accompanying drawings.

43. Software substantially as herein described and with reference to the accompanying drawings.

Application No: GB 9930699.5
Claims searched: 1-16

Examiner: Gareth Griffiths
Date of search: 22 June 2000

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): H4P (PPA, PPEC)

Int Cl (Ed.7): G06F 13/14, 13/38, 17/30, H04L 29/06

Other: Online Databases: WPI, EPODOC, JAPIO

Documents considered to be relevant:

Category	Identity of document and relevant passage		Relevant to claims
X, E	EP0969389 A2	(IBM) whole document	1 at least
A	EP0718783 A1	(SUN MICROSYSTEMS) whole document	
X, P	WO99/61984 A1	(3COM) see esp. pages 16, 19 & 20	1 at least
X, P	WO99/35595 A2	(SONERA) whole document	1 at least
X	WO98/36344 A2	(LEXTRON) p.9 line 6 - p.10 line 31	1 at least

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

THIS PAGE BLANK (USPTO)

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☑ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

THIS PAGE BLANK (USPTO)